

DN-PCI-5110

DeviceNet 主站卡

DS01020405 Rev 1.22 Date:2007/07/20 产品数据手册

概述

DN-PCI-5110 DeviceNet 主站卡采用 PCI 接口，使 PC 连接 CAN 总线并实现 DeviceNet 协议或 CAN2.0B 协议的通讯。

DN-PCI-5110 主站卡的 CAN 通道集成有完全的电气隔离保护电路。

DN-PCI-5110 主站卡提供 DeviceNet 扫描模块、程序实例、测试软件，支持配置并检测 DeviceNe 网络。

DN-PCI-5110 主站卡支持 Win98/Me、Win2000/XP 操作系统，提供 OPC 服务器，可运行在组态软件中。

产品特性

- 通用 PCI 接口，兼容 PCI2.2 规范
- 通讯协议符合 DeviceNet 规范 V2.0 版
- 同时支持 CAN2.0A 和 CAN2.0B 规范
- 支持 125、250、500kbps 共 3 种通讯速率
- 采用 5 针开放型 DeviceNet 插座
- 外部电源采用 9~24V 输入电源
- DC 2500V 电气隔离保护
- 提供 DNSCAN 扫描模块
- 提供 OPC 服务器
- 遵守工业应用规范

操作系统支持

DN-PCI-5110 主站卡支持 Win98/Me、Win2000/XP 操作系统，提供 OPC 服务器，可运行在组态软件中。

如果客户有特殊要求，请与广州致远电子有限公司联系。

订购信息

型号	工作温度	接口
DN-PCI-5110	0°C ~ +70°C	OPEN5

规格

硬件

CAN 控制器：SJA1000T

CAN 收发器：PCA82C251T

隔离：DC2500V

接口

总线：PCI ver. 2.2 (32-bit)

性能

速率：125Kbps、250Kbps、500Kbps

接口

OPEN5 开放式 DeviceNet 连接器

电源

需外加+9~24V 电源

操作系统支持

Win98/Me、Win2000、WinXP

工具软件支持

DNSCAN 模块

DNSCAN 示范例程

OPC 服务器 ZOPC_Server

电源和环境

电源要求：5V@300 mA (Max.)

操作温度：0°C ~ +70°C

存储温度：-40°C ~ +85°C

尺寸：130 x 90 mm (W x D)



修订历史

版本	日期	原因
V1.00	2004/10/30	建立文档
V1.10	2007/07/20	修改文档的格式成统一版本

目录

1. 功能简介

DN-PCI-5110 DeviceNet 主站卡是具有 PCI 接口的高性能 CAN 总线通讯适配卡，使 PC 机方便地连接到 CAN 总线上，能够实现 DeviceNet 协议或 CAN2.0B 协议的数据通讯。

DN-PCI-5110 主站卡采用标准 PCI 接口，集成 8KB 高速缓冲区实现与主机 PC 的数据交换。接口卡采用 4 层电路板，SMD 工艺，完全电气隔离，使 PC 机避免由于地环流的损坏，增强系统在恶劣环境中使用的可靠性。

DN-PCI-5110 主站卡提供 DeviceNet 扫描模块动态链接库，可以作为网络中的 DeviceNe 主站卡运行；配套的程序示例、测试软件可以方便地对 DeviceNe 网络进行配置并检测其运行状态。接口函数部分详细地列出了关于 DeviceNet 扫描模块动态链接库的应用信息。

DN-PCI-5110 主站卡可运行于 Win98/Me、Win2000/XP 操作系统下，具有 ZOPC 服务器接口，可以直接运行在通常使用的组态软件下，比如组态王、MCGS、Intouch 等软件。

1.1 产品外观

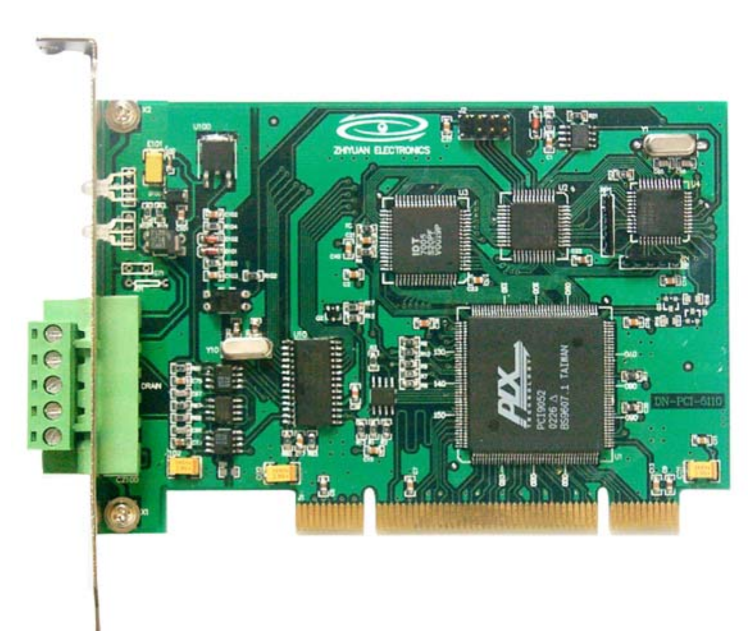


图 1-1　DN-PCI-5110　DeviceNet 主站卡

1.2 性能规格

- PC 接口：　32 位 33M PCI 数据总线，PCI2.2 兼容，即插即用；
- 板载数据存储器：　8KB　DPRAM，由高速微处理器控制；
- CAN控制器：　NXP　SJA1000T；
- CAN收发器：　NXP　PCA82C251；
- 数据传送速率：　CAN 通讯速率可编程，范围在 5Kbit/s～1Mbit/s 内；
- DeviceNet 通讯速率：　支持 125kbps、250kbps、500kbps 标准 DeviceNet 通讯速率；
- DeviceNet 连接器：　符合 DeviceNet 规范 V2.0 的 5 针开放型插座；
- DeviceNet 输入电源：　符合 DeviceNet 标准，采用外部 9-24V 输入电源；
- 通讯协议　符合 DeviceNet 协议规范 V2.0 版；

- 光电隔离耐压：　　2500VDC；
- 工作环境温度：　　0℃～70℃；
- 物理尺寸：　　标准 PCI 短卡（130mm * 90mm）；
- 运行环境：　　Win98/Me、Win2000/XP 操作系统；

1.3 软件支持

DN-PCI-5110 DeviceNet 主站卡采用 WDM 驱动库，支持 Win98/Me、Win2000/XP 操作系统，支持一机多卡，可以实现 DeviceNet 协议和 CAN2.0B 协议的数据通讯。

DN-PCI-5110 DeviceNet 主站卡提供方便使用、功能完善的 DeviceNet 接口函数库文件，用户可以捆绑 DN-PCI-5110 主站卡自由发放相应的驱动程序及应用程序文件，或直接作为 DeviceNet 主站卡应用在工程项目中。

1.4 产品清单

- DN-PCI-5110 DeviceNet 主站卡　　1 块
- DN-PCI-5110 DeviceNet 主站卡数据手册　　1 份
- DeviceNet 接口函数库与测试软件　　1 份
- 测试软件、OPC 服务器软件　　1 份
- CAN-bus/DeviceNet 设计资料　　1 份

2. 设备安装

2.1 安装硬件

DN-PCI-5110 DeviceNet 主站卡属于静电敏感产品，出厂时安放在专用保护袋中。因此，在对接口卡进行操作时，请注意采取必要的防护措施，以保证接口卡不受损坏。

硬件安装时要在 PC 断电状态下，同样，拆卸 DN-PCI-5110 DeviceNet 主站卡也应当在 PC 断电的状态下进行。

DN-PCI-5110 DeviceNet 主站卡没有任何开关和跳线用于分配中断和 I/O 地址，这些都是由 BIOS 自动分配的。因此，在安装驱动程序之前板卡必须事先安装到 PCI 槽上。以下是安装步骤：

1. 关闭 PC 电源。
2. 打开 PC 的盖子。
3. 将 DN-PCI-5110 DeviceNet 主站卡插入空闲的 PCI 插槽。
4. 拧紧固定板卡的螺钉。
5. 打开 PC 电源，此时 BIOS 会自动给 DN-PCI-5110 DeviceNet 主站卡分配中断和 I/O 地址。

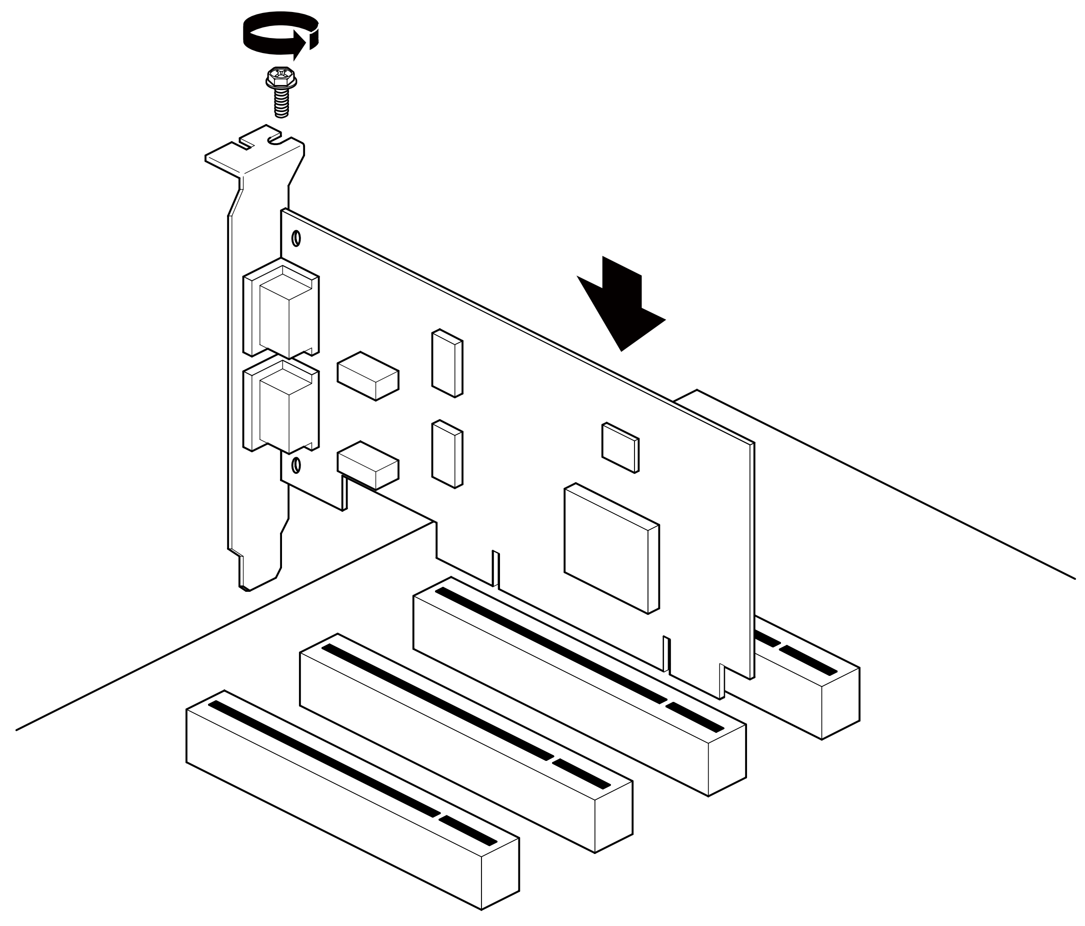


图 2-1　安装 DN-PCI-5110 DeviceNet 主站卡

注意事项：
请勿带电插拔 PCI 接口卡；安装时不要用手触摸器件，防止静电损坏器件。

2.2 DeviceNet 连接器与指示灯

2.2.1 DeviceNet 连接器

DN-PCI-5110 DeviceNet 主站卡上集成 1 个 5 针开放式插座，作为 DeviceNet 连接器，其规格符合 DeviceNet 规范 V2.0 的要求。关于 DeviceNet 连接器的信号定义如表格 2-1 所示；外形与标示如图 2-1 所示。

表格 2-1　DeviceNet 连接器的信号定义

引脚号	信号	颜色	功能
1	V-	黑	DeviceNet 输入电源负极
2	CANL	蓝	CANL 信号线
3	SHIELD	灰	屏蔽线
4	CANH	白	CANH 信号线
5	V+	红	DeviceNet 输入电源正极

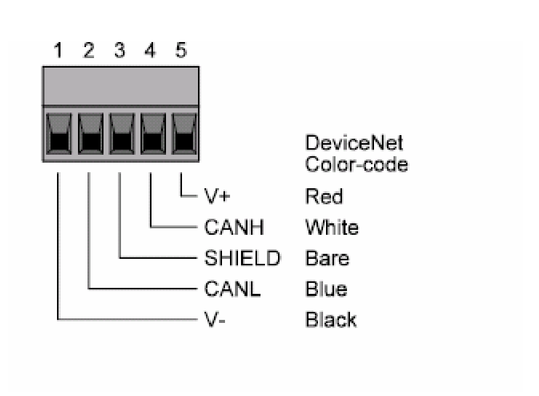


图 2-2　DeviceNet 连接器所用的 5 针开放式插座

2.2.2 输入电源

DN-PCI-5110 DeviceNet 主站卡采用由总线向 CAN 收发器电路供电方式，必须由外部向接口卡提供 9～24V DC 输入电源，接口卡上集成电源调节器、反接保护电路。电路符合 DeviceNet 规范 V2.0 的要求。

2.2.3 指示灯

DN-PCI-5110 DeviceNet 主站卡提供 2 个双色指示灯：MS（模块状态指示）、NS（网络状态指示）。这 2 个指示灯的状态符合 DeviceNet 规范中的定义（见 DeviceNet 规范 I 第 8 章）。

2.2.4 终端电阻

为了增强 CAN 通讯的可靠性，CAN 总线网络的两个端点通常要加入 120Ω 终端匹配电阻，终端匹配电阻并联在 CANH 信号、CANL 信号之间，如图 2-3 所示。

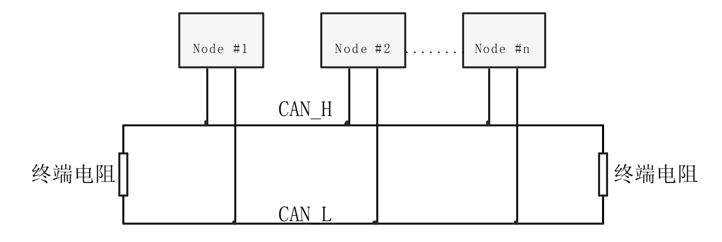


图 2-3 **CAN 网络结构**

当 DN-PCI-5110 DeviceNet 主站卡不处于 DeviceNet 网络或 CAN 网络的终端位置时，120Ω终端匹配电阻可以不用安装。

3. 驱动的安装

请将配套光盘<Driver>目录的文件拷贝到硬盘。为了确保任何时候安装都可以正确指定相应的驱动程序，请严格按照以下步骤进行安装处理。

注意：DN-PCI-5110 DeviceNet 主站卡使用的是 PCI-51XX 系列驱动程序。

3.1 在 WinXP 系统下安装

如果已将 DN-PCI-5110 主站卡插入 PC 的 PCI 插槽，则在重启系统之后，PC 会提示发现新硬件，如图 3-1 所示，此时应该选择“从列表或指定位置安装（高级）”，然后单击“下一步”：



图 3-1 欢迎使用找到新硬件向导

当出现如图 3-2 所示的对话框，选择“不要搜索，我要自己选择要安装的驱动程序”项，然后单击“下一步”：

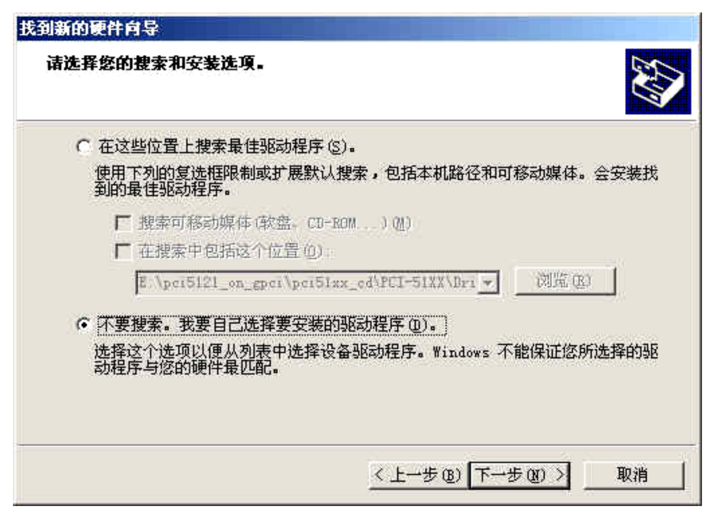


图 3-2 请选择你的搜索和安装选项

当出现如图 3-3 所示对话框时，单击“型号” 栏中的空白处，暂时不选中任何型号，接着单击“从磁盘安装”按钮以指定 DN-PCI-5110 主站卡的安装目录位置。



图 3-3 选择要为此硬件安装的设备驱动程序（1）

当出现如图 3-4 所示对话框时，我们可以通过"浏览"按键找到驱动程序的 inf 文件：

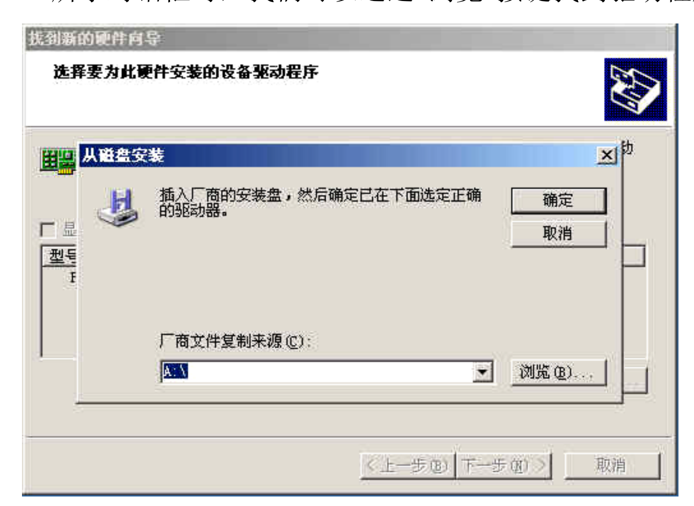


图 3-4 选择"从磁盘安装"

查找文件结果如图 3-5 所示，选中相应文件"PCI51XX.inf"后单击"打开"：

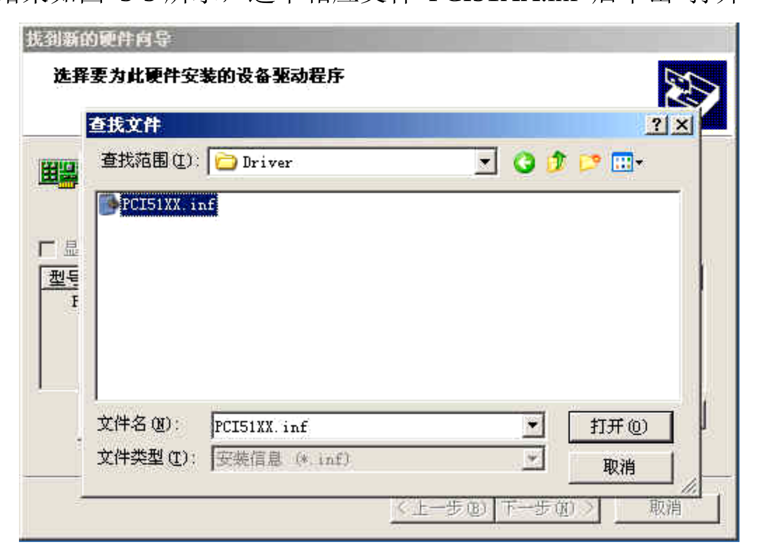


图 3-5 查找文件（1）

当出现如图 3-6 对话框并确定目录正确后（目录不对的话必须单击"上一步"重新查找到对为止），单击"确定" 键：



图 3-6 查找文件（2）

接着如图 3-7 所示，这时我们才选中相应的板卡型号，确定选中正确型号后单击"下一步"：

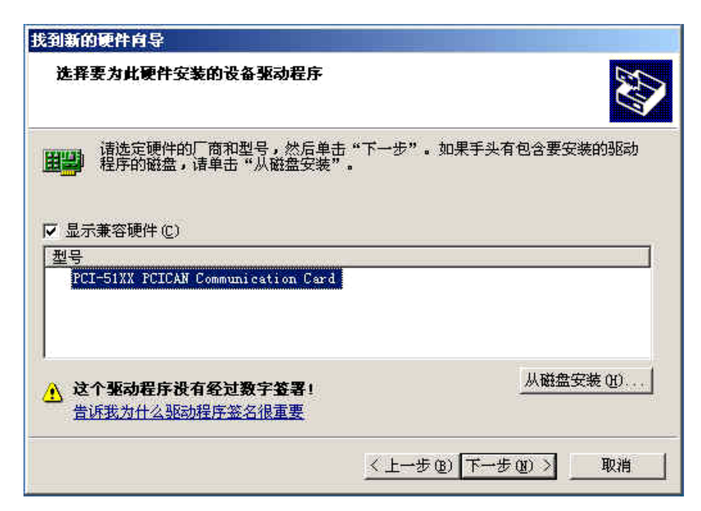


图 3-7 选择要为此硬件安装的设备驱动程序（2）

继续安装，直到出现图 3-8 所示对话框；此时，单击"完成"，即完成了驱动的安装。



图 3-8 完成找到新硬件向导

安装成功后，"设备管理器"中将列出所安装的 DN-PCI-5110 主站卡。图 3-9 为安装完成 1 块 DN-PCI-5110 主站卡后的设备管理器界面。

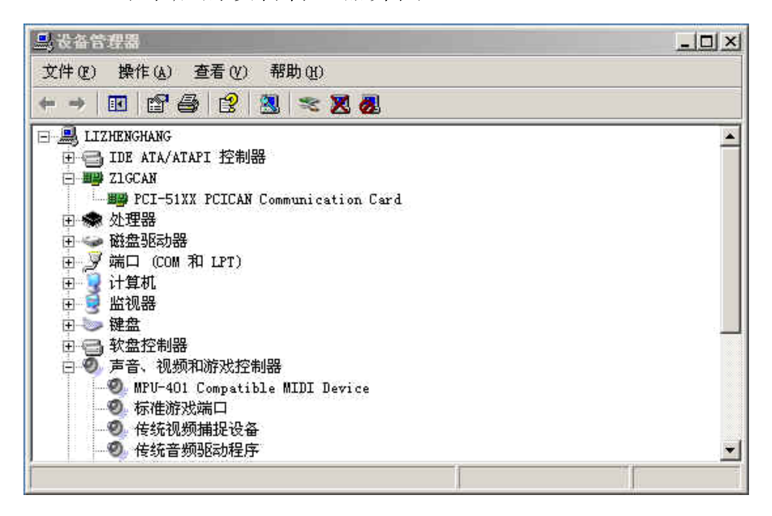


图 3-9 **Win XP 的设备管理器**

3.2 在 Win2000 系统下安装

在 Win2000 系统下安装 DN-PCI-5110 主站卡的过程与 WinXP 系统类似，这里不再多讲，需要注意的是：任何时候安装都要尽可能手工选择，而不要使用自动安装。

3.3 在 Win98/Me 系统下安装

以在 Win98 下安装为例子，进入"设备管理器"，如图 3-10 所示。



图 3-10 **Win98 的设备管理器（1）**

单击"刷新"，启动硬件安装向导，弹出如图 3-11 所示的对话框，此时对话框上显示发现了"PCI-51XX Communication Card"：

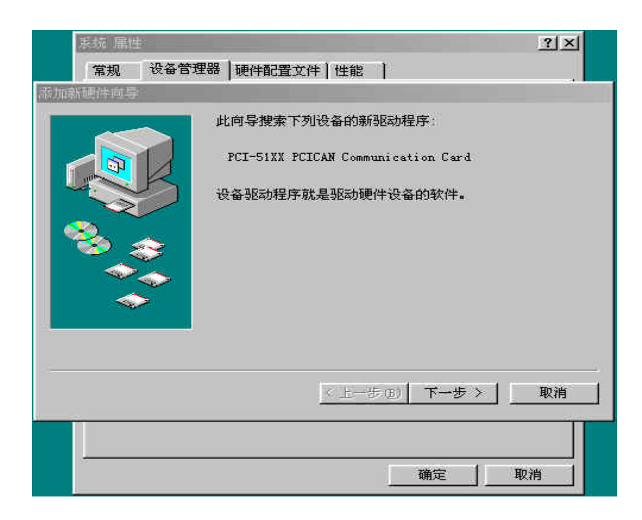


图 3-11　添加新硬件向导（1）

单击“下一步”后出现如下对话框：

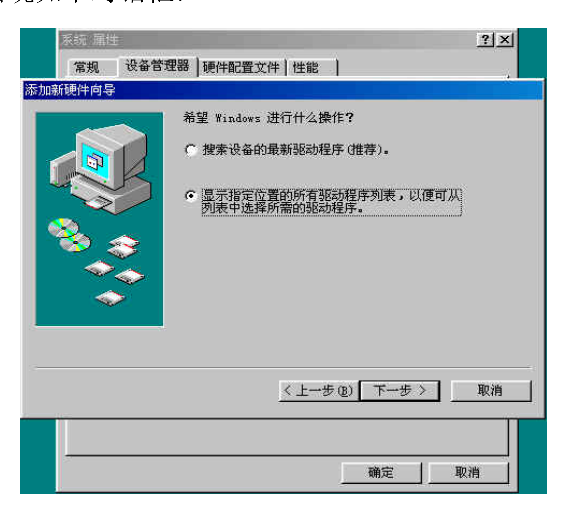


图 3-12　添加新硬件向导（2）

选择“显示指定位置的所有驱动程序列表，以便可从列表中选择所需的驱动程序”，然后单击“下一步”，出现图 3-13 所示的对话框：

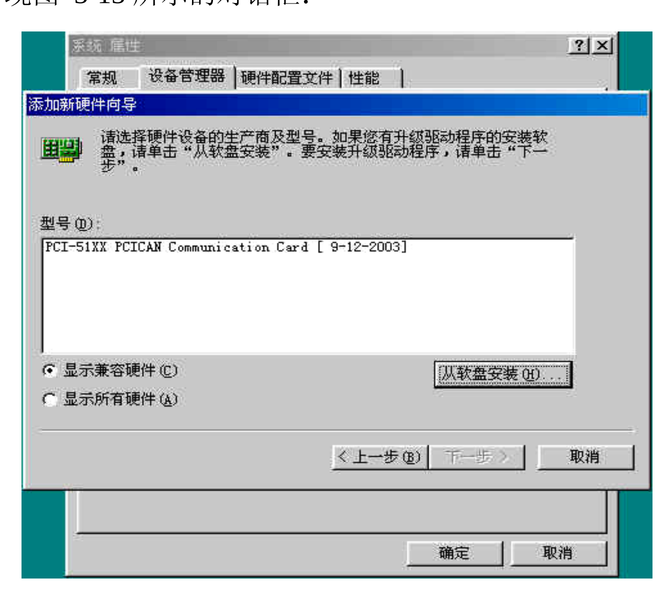


图 3-13　添加新硬件向导（3）

在“型号”栏的空白处单击一下，暂时不选中任何型号，接着单击“从磁盘安装”按钮以指定 DN-PCI-5110 主站卡的安装目录位置。当出现如图 3-14 所示的对话框时：

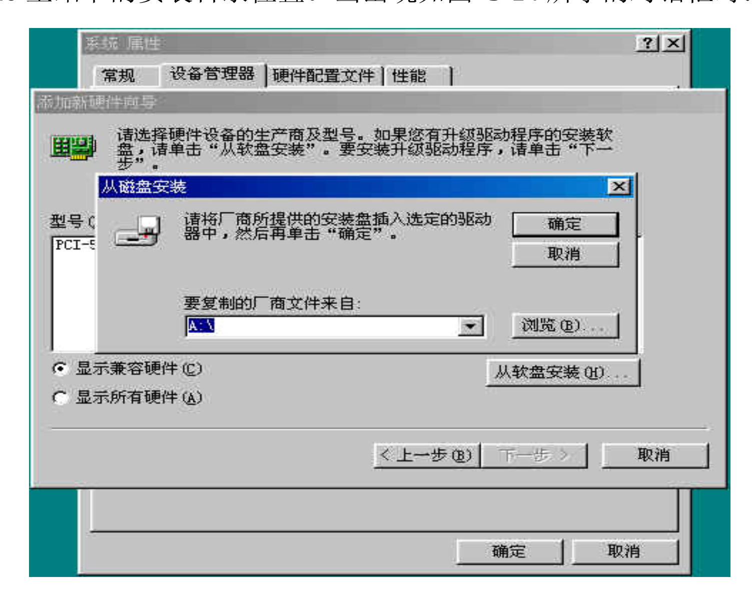


图 3-14　添加新硬件向导（4）

单击“浏览”寻找相应的 inf 文件，如图 3-15 所示：

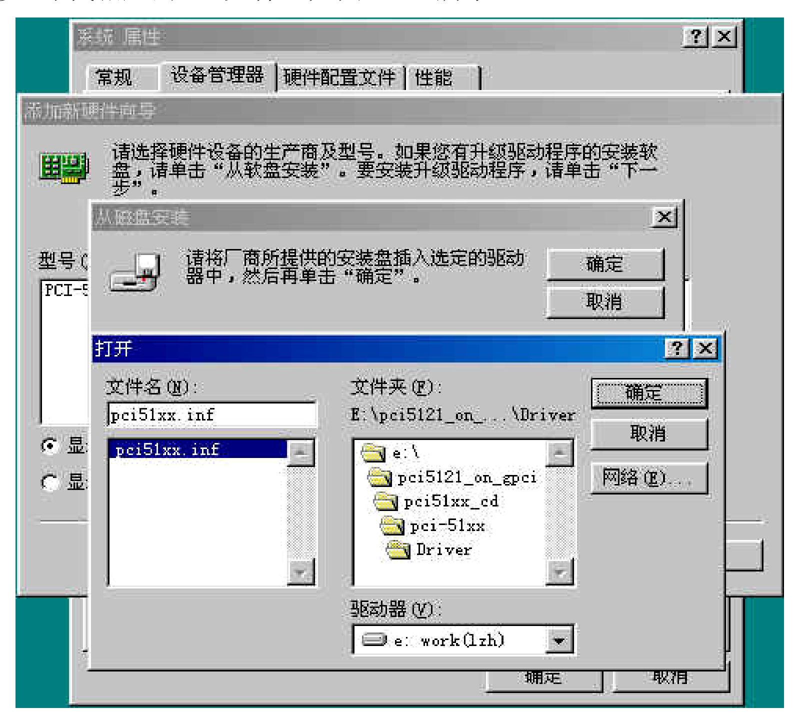


图 3-15　添加新硬件向导（5）

选中所要的文件后，分别单击两个“确定”键，进入图 3-16 所示的界面：:

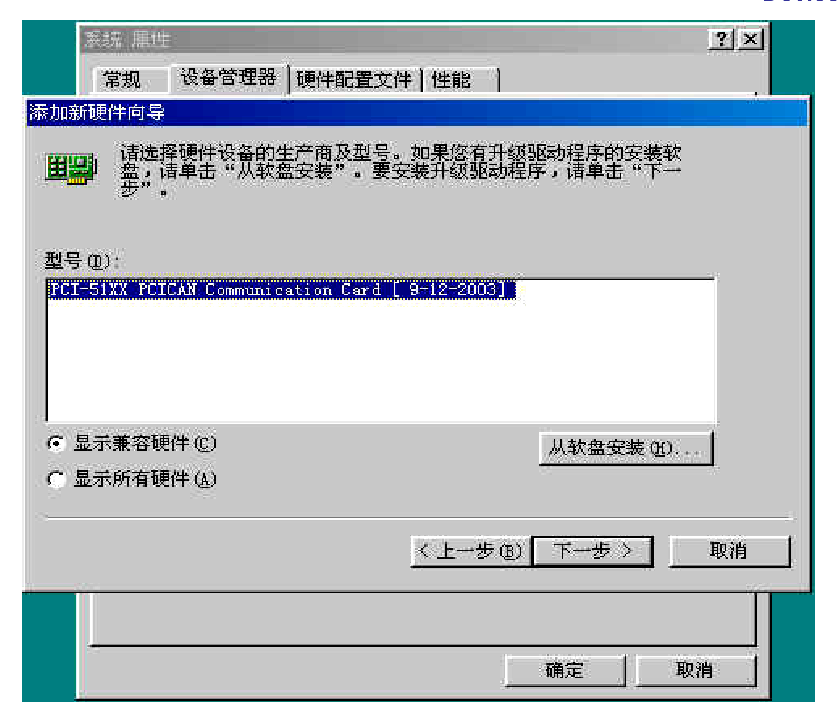


图 3-16　**添加新硬件向导（6）**

这个时候才要选中相应的型号，然后单击"下一步"，当出现图 3-17 所示的界面时：

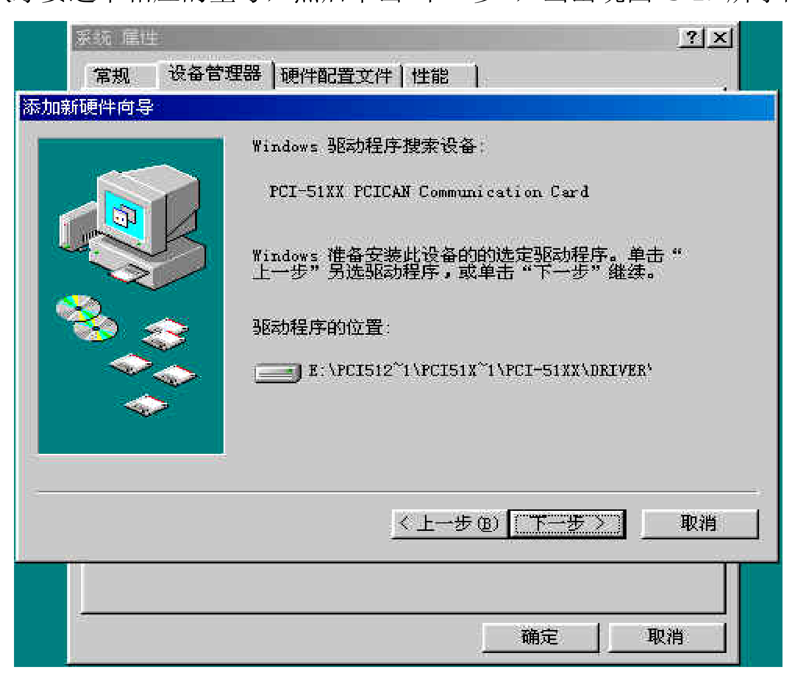


图 3-17　**添加新硬件向导（7）**

确定所选的驱动文件位置正确后（目录不对的话必须单击"上一步"重新查找到对为止），单击"下一步"，开始安装驱动程序。在安装过程中如果未发生任何错误，最后将进入图 3-18 所示的界面，单击"完成"就可以了。



图 3-18 **添加新硬件向导（8）**

安装完成后"设备管理器"将出现如图 3-19 的选项，表示驱动已经安装成功。

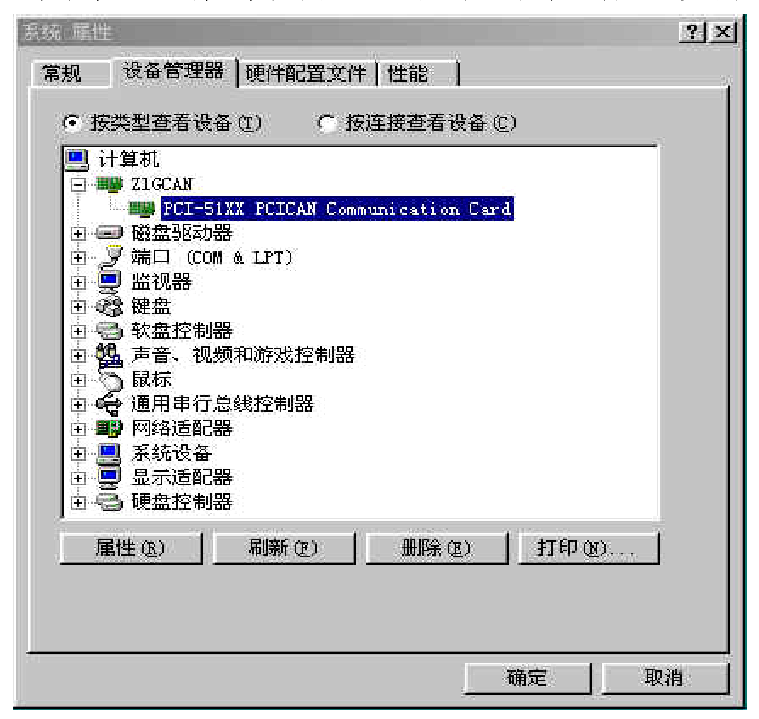


图 3-19 **Win98 的设备管理器（2）**

4. 接口函数

DN-PCI-5110 主站卡采用 WDM 驱动程序，支持 Win98/Me、Win2000、WinXP 操作系统，支持一机多卡；同时提供完整应用示范代码，包含 C++Builder 等开发例程示范，方便用户进行二次开发。

4.1 主题

接口函数描述基于 DN-PCI-5110 主站卡的 DNSCAN 模块。DNSCAN 模块是 DeviceNet 网络中的一个节点，该节点管理 DeviceNet 上其他设备和主机（PC）的数据交换，并且可以具有主站和从站的双重身份。

DeviceNet主站卡：

- 不需要主机的干涉而自动管理网络上大部分的数据交换
- 对出入主机的通信数据提供高速内存映射接口

公共通信能力：

- 支持 UCMM，支持动态的显式信息交换
- 外部设备可以通过 DeviceNet 网络访问主机内的应用对象

主站（组2客户机）能力：

- 支持显式信息通信
- 支持具有仅限组 2 从站能力和具有 UCMM 功能的设备
- 自动校验设备的身份标识
- 自动重连超时或连接失败的从站

从站（组2服务器）能力：

- 支持轮询、选通或状态改变/周期 I/O 信息
- 支持显式信息通信

4.2 DeviceNet 相关描述

本节内容介绍 DeviceNet 主站模块特定的对象模型，通过阅读 DeviceNet 规范可以获得对象模型的相关背景知识。

注意：下面表格中的空心小圈表示该定义可选或没有。

4.2.1 一致性测试

DNSCAN 暂时还没有执行 ODVA 协会的一致性测认证；但产品已经通过实际 DeviceNet 网络的运行测试，可以在 DeviceNet 网络可靠运行。

4.2.2 版本声明

DNSCAN 兼容 DeviceNet 规范：卷 I 2.0 和卷 II 2.0。

4.2.3 通信能力

1. 传输速率

- 125 Kbps，250 Kbps 和 500 Kbps

2. 预定义主/从连接组

- 支持组 2 客户机（主站）
- 支持仅限组 2 客户机（主站）
- 支持组 2 服务器（从站）

3. 动态连接

- 支持 UCMM
- 动态连接可使用组 1 信息 ID、组 2 信息 ID 和组 3 信息 ID

4. 传输类型

- 显式信息
- I/O 信息

4.2.4 标识对象

类代码：0x01

该类仅有一个实例。

1. 类属性

表格 4-1 标识对象类属性

ID	Description	Get	Set	Limits
1	Revision	○	○	
2	Max Instance	○	○	
6	Max ID of class attributes	○	○	
7	Max ID of instance attributes	○	○	
●Supported ○Not supported				

2. 类服务

表格 4-2 标识对象类服务

Service		Param Options
Get_Attributes_All	○	
Reset	○	
Get_Attribute_Single	○	
Find_Next_Object_Instance	○	
●Supported ○Not supported		

3. 实例属性

表格 4-3 标识对象实例属性

Id	Description	Get	Set	Limits
1	Vendor	●	○	0333hex
2	Device Type	●	○	0Chex
3	Product Code	●	○	
4	Revision [1]	●	○	[1,100]
5	Status	●	○	
6	Serial Number	●	○	
7	Product Name	●	○	

8	State	○	○	
●Supported ○Not supported				
*1 DeviceNet 规范规定主、次版本都不能为0，最低此版本为1。				

4. 实例服务

表格 4-4 标识对象实例服务

Service		Param Options
Reset	●	
Get_Attribute_Single	●	
●Supported ○Not supported		

4.2.5 消息路由对象

类代码：0x02

该类仅有一个实例。

1. 类属性

表格 4-5 消息路由对象类属性

Id	Description	Get	Set	Limits
1	Revision	○	○	
4	Optional Attribute List	○	○	
5	Optional Service List	○	○	
6	Max ID of class attributes	○	○	
7	Max ID of instance attributes	○	○	
●Supported ○Not supported				

2. 类服务

表格 4-6 消息路由对象类服务

Service		Param Options
Get_Attributes_All	○	
Get_Attribute_Single	○	
●Supported ○Not supported		

3. 实例属性

表格 4-7 消息路由对象实例属性

Id	Description	Get	Set	Limits
1	Object List	○	○	
2	Maximum connections supported	○	○	
3	Number of active connections	○	○	
4	Active connections list	○	○	
●Supported ○Not supported				

4. 实例服务

表格 4-8 消息路由对象实例服务

Service		Param Options
Get_Attributes_All	○	
Get_Attribute_Single	○	
●Supported ○Not supported		

4.2.6 DeviceNet 对象

类代码：0x03

该类仅有一个实例。

1. 类属性

表格 4-9 DeviceNet 对象类属性

Id	Description	Get	Set	Limits
1	Revision	●	○	
●Supported ○Not supported				

2. 类服务

表格 4-10 DeviceNet 对象类服务

Service		Param Options
Get_Attribute_Single	●	
●Supported ○Not supported		

3. 实例属性

表格 4-11 DeviceNet 对象实例属性

Id	Description	Get	Set	Limits
1	MAC ID	●	○	
2	Baud Rate	●	○	
3	BOI	○	○	
4	Bus-off counter	○	○	
5	Allocation information	●	○	
6	MAC ID switch changed	○	○	
7	Baud rate switch changed	○	○	
8	MAC ID switch value	○	○	
9	Baud rate switch value	○	○	
●Supported ○Not supported				

4. 实例服务

表格 4-12 DeviceNet 对象实例服务

Service		Param Options
Get_Attribute_Single	●	
Allocate M/S connection set	●	
Release M/S connection set	●	
●Supported ○Not supported		

4.2.7 连接对象

类代码：0x05

两种连接类型（显式、I/O）都可以有最大 255 个类实例。

1. 类属性

表格 4-13 连接对象类属性

Id	Description	Get	Set	Limits
1	Revision	○	○	
●Supported ○Not supported				

2. 类服务

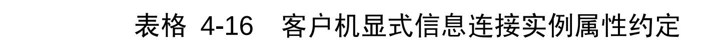
表格 4-14 连接对象类服务

Service		Param Options
Reset	○	
Create	○	
Delete	○	
Get_Attribute_Single	○	
Find_Next_Object_Instance	○	
●Supported ○Not supported		

3. 实例属性

表格 4-15 连接对象实例属性

Id	Description	Get	Set	Limits *1
1	State	●	○	
2	Instance Type	●	○	
3	Transport class trigger	●	○	
4	Produced connection ID	●	○	
5	Consumed connection ID	●	○	
6	Initial comm. characteristics	●	○	
7	Produced connection size	●	○	
8	Consumed connection size	●	○	
9	Expected packet rate	●	●	
12	Watchdog timeout action 2	●	●	
13	Produced connection path length	●	○	
14	Produced connection path	●	○	
15	Consumed connection path length	●	○	
16	Consumed connection path	●	○	
17	Production Inhibit Time	●	●	
●Supported ○Not supported *1 具体连接类型的参数限制将被列于下面的表格中 2 该属性只能通过显式信息连接进行设置				



表格 4-16 客户机显式信息连接实例属性约定

Id	Description	Limits/Fixed Value
1	State	
2	Instance Type	0
3	Transport class trigger	0x23
4	Produced connection ID	
5	Consumed connection ID	
6	Initial comm. characteristics	
7	Produced connection size	0xffff
8	Consumed connection size	0xffff
9	Expected packet rate	2500
12	Watchdog timeout action	0x01
13	Produced connection path length	0
14	Produced connection path	—
15	Consumed connection path length	0
16	Consumed connection path	—
17	Production Inhibit Time	—

表格 4-17 服务器显式信息连接实例属性约定

Id	Description	Limits/Fixed Value
1	State	
2	Instance Type	0
3	Transport class trigger	0x83
4	Produced connection ID	
5	Consumed connection ID	
6	Initial comm. characteristics	
7	Produced connection size	0xffff
8	Consumed connection size	0xffff
9	Expected packet rate	2500
12	Watchdog timeout action	0x01 or 0x03
13	Produced connection path length	0
14	Produced connection path	—
15	Consumed connection path length	0
16	Consumed connection path	—
17	Production Inhibit Time	0xffff

表格 4-18 客户机轮询 I/O 连接属性约定

Id	Description	Limits/Fixed Value
1	State	
2	Instance Type	1
3	Transport class trigger	0x22
4	Produced connection ID	
5	Consumed connection ID	

6	Initial comm. characteristics	0x10
7	Produced connection size	
8	Consumed connection size	
9	Expected packet rate	250
12	Watchdog timeout action	0
13	Produced connection path length	
14	Produced connection path	
15	Consumed connection path length	
16	Consumed connection path	
17	Production Inhibit Time	—

表格 4-19 服务器轮询 I/O 连接属性约定

Id	Description	Limits/Fixed Value
1	State	
2	Instance Type	1
3	Transport class trigger	0x83
4	Produced connection ID	
5	Consumed connection ID	
6	Initial comm. characteristics	0x01
7	Produced connection size	
8	Consumed connection size	
9	Expected packet rate	
12	Watchdog timeout action	0
13	Produced connection path length	
14	Produced connection path	
15	Consumed connection path length	
16	Consumed connection path	
17	Production Inhibit Time	—

表格 4-20 客户机选通 I/O 连接属性约定

Id	Description	Limits/Fixed Value
1	State	
2	Instance Type	1
3	Transport class trigger	0x80
4	Produced connection ID	
5	Consumed connection ID	
6	Initial comm. characteristics	0xf0
7	Produced connection size	
8	Consumed connection size	
9	Expected packet rate	250
12	Watchdog timeout action	0
13	Produced connection path length	
14	Produced connection path	

15	Consumed connection path length	
16	Consumed connection path	
17	Production Inhibit Time	—

表格 4-21　服务器选通 I/O 连接属性约定

Id	Description	Limits/Fixed Value
1	State	
2	Instance Type	1
3	Transport class trigger	0x83
4	Produced connection ID	
5	Consumed connection ID	
6	Initial comm. characteristics	0x02
7	Produced connection size	
8	Consumed connection size	
9	Expected packet rate	
12	Watchdog timeout action	0
13	Produced connection path length	
14	Produced connection path	
15	Consumed connection path length	
16	Consumed connection path	
17	Production Inhibit Time	—

表格 4-22　客户机状态改变/周期 I/O 连接属性约定（有应答）

Id	Description	Limits/Fixed Value
1	State	
2	Instance Type	1
3	Transport class trigger	0x03 or 0x13
4	Produced connection ID	
5	Consumed connection ID	
6	Initial comm. characteristics	0x01
7	Produced connection size	
8	Consumed connection size	
9	Expected packet rate	250
12	Watchdog timeout action	0
13	Produced connection path length	
14	Produced connection path	
15	Consumed connection path length	
16	Consumed connection path	
17	Production Inhibit Time	—

表格 4-23　服务器状态改变/周期 I/O 连接属性约定（有应答）

Id	Description	Limits/Fixed Value
1	State	
2	Instance Type	1

3	Transport class trigger	0x03 or 0x13
4	Produced connection ID	
5	Consumed connection ID	
6	Initial comm. Characteristics	0x01
7	Produced connection size	
8	Consumed connection size	
9	Expected packet rate	250
12	Watchdog timeout action	0
13	Produced connection path length	
14	Produced connection path	
15	Consumed connection path length	4
16	Consumed connection path	0x20 0x2B 0x24 0x01
17	Production Inhibit Time	—

表格 4-24　客户机状态改变/周期 I/O 连接属性约定（无应答）

Id	Description	Limits/Fixed Value
1	State	
2	Instance Type	1
3	Transport class trigger	0x00 or 0x10
4	Produced connection ID	
5	Consumed connection ID	0xffff
6	Initial comm. characteristics	0x0f
7	Produced connection size	
8	Consumed connection size	
9	Expected packet rate	250
12	Watchdog timeout action	0
13	Produced connection path length	
14	Produced connection path	
15	Consumed connection path length	0
16	Consumed connection path	—
17	Production Inhibit Time	—

表格 4-25　服务器状态改变/周期 I/O 连接属性约定（无应答）

Id	Description	Limits/Fixed Value
1	State	
2	Instance Type	1
3	Transport class trigger	0x00 or 0x10
4	Produced connection ID	
5	Consumed connection ID	0xffff
6	Initial comm. Characteristics	0x0f
7	Produced connection size	
8	Consumed connection size	
9	Expected packet rate	250

12	Watchdog timeout action	0
13	Produced connection path length	
14	Produced connection path	
15	Consumed connection path length	0
16	Consumed connection path	—
17	Production Inhibit Time	—

4. 实例服务

表格 4-26 连接对象实例服务

Service		Param Options
Reset	○	
Delete	○	
Apply_Attributes	○	
Get_Attribute_Single	●	
Set_Attribute_Single	●	
●Supported ○Not supported		

4.3 配置说明

4.3.1 简介

本节描述 DN-PCI-5110 主站卡为实现 DeviceNet 网络功能所必需的配置任务和背景知识。通过配置，DNSCAN 模块可以实现下列功能：

- DeviceNet 客户机：
  （1） 支持最多 63 个服务器（从站）设备
  （2） 设备扫描列表可以被动态修改
  （3） 支持仅限组 2 从站
  （4） 支持组 2 从站（有 UCMM 功能）
  （5） 支持具有 UCMM 功能的服务器能力（仅限于显式信息通信）
- DeviceNet 服务器：
  （1） 具有 UCMM 能力的组 2 服务器
  （2） 支持选通、轮询、状态改变/周期 I/O 信息连接
  （3） 主机中的对象可以被外部设备通过 DeviceNet 网络进行访问

4.3.2 专业术语

因为 DNSCAN 模块同时支持 DeviceNet 客户机和服务器能力，所以必须遵循一定的规则来避免冲突。

输入/输出

输入/输出都以处理器为参考点，输入数据是指从现场总线过来，传往处理器的数据，输出数据是指源于处理器，发往现场总线的数据。

生产者和消费者

DeviceNet 是一种基于连接的网络，所有的通信必须通过连接来完成。一个连接就是一条连通生产者（产生信息者）和消费者（接收信息者）的路径，当连接处在发送数据时，可以认为它在生产数据；当连接处在接收数据时，可以认为它在消费数据。

客户机/服务器

客户机通过连接发出通信数据，服务器通过连接对通信数据进行处理，并在需要时返回应答信息。

主站 / 从站角色

主站在组 2 主/从连接中充当客户机节点，主站向从站传输输出数据，并接收从站传来的输入数据，主站还向从站发送显式请求信息。

从站在组 2 主/从连接中充当服务器节点，从站接收输出数据，并向主站传输输入数据，从站接收并响应主站传来的显式信息。

4.3.3 主机接口

主机和 DNSCAN 使用共享内存上的主机接口块进行数据交换，当共享内存上的数据发生变化时，主机和 DNSCAN 都可以向对方产生中断信号。

4.3.4 服务器配置

DNSCAN 有能力充当组 2 服务器（从站）和显式信息服务器（借助于 UCMM），以下段落将描述和这种能力相关的配置参数。

1. 服务器连接配置

这些参数规定了关于服务器通信的类型和各种连接的配置需求。

表格 4-27 服务器连接配置参数

Item	Description
ConnectionFlags	连接组合标志字，指定服务器所支持的连接类型，最多支持 2 个 I/O 连接和 1 个显式信息连接
ExplicitRequestSize	显式请求信息缓冲长度，如果不使用显式连接就忽略该参数
ExplicitResponseSize	显式应答信息缓冲长度，如果不使用显式连接就忽略该参数
Output1Size	第一个 I/O 连接的生产连接长度
Input1Size	第一个 I/O 连接的消费连接长度
Output2Size	第二个 I/O 连接的生产连接长度
Input2Size	第二个 I/O 连接的消费连接长度

2. ConnectionFlags

此参数指定了应用所支持的连接组合类型。

表格 4-28 服务器连接组合类型参数位定义

Offset	Bit							
	7	6	5	4	3	2	1	0
0	Res.	AKS	CYC	COS	Res.	ST	P	EX
	Reserved							

阴影部分为以后保留

仅限组 2 I/O 连接标志

AKS 应答抑制位

CYC 循环I/O连接

COS 状态改变I/O连接

ST 位选通I/O连接

P 轮询I/O连接

显式信息连接标志

EX 显式信息连接，组 2 预定义和组 3 动态显式信息连接

3. 所支持的连接标志组合

下表列出了所有的连接组合类型。

表格 4-29 服务器连接标志组合列表

Flags	Explicit	I/O1	I/O2
0x0001	G3 Dynamic [2]	N/A	N/A
0x0002	NO	G2 M/S Poll	N/A
0x0003	G2 M/S or G3 Dynamic[1]	G2 M/S Poll	N/A
0x0004	NO	G2 M/S Strobe	N/A
0x0005	G2 M/S or G3 Dynamic[1]	G2 M/S Strobe	N/A
0x0006	NO	G2 M/S Poll	G2 M/S Strobe
0x0007	G2 M/S or G3 Dynamic[1]	G2 M/S Poll	G2 M/S Strobe
0x0010	NO	G2 M/S COS	N/A
0x0011	G2 M/S or G3 Dynamic[1]	G2 M/S COS	N/A
0x0012	NO	G2 M/S Poll	G2 M/S COS
0x0013	G2 M/S or G3 Dynamic[1]	G2 M/S Poll	G2 M/S COS
0x0014	NO	G2 M/S Strobe	G2 M/S COS
0x0015	G2 M/S or G3 Dynamic[1]	G2 M/S Strobe	G2 M/S COS
0x0020	NO	G2 M/S Cyclic	N/A
0x0021	G2 M/S or G3 Dynamic[1]	G2 M/S Cyclic	N/A
0x0022	NO	G2 M/S Poll	G2 M/S Cyclic
0x0023	G2 M/S or G3 Dynamic[1]	G2 M/S Poll	G2 M/S Cyclic
0x0024	NO	G2 M/S Strobe	G2 M/S Cyclic
0x0025	G2 M/S or G3 Dynamic[1]	G2 M/S Strobe	G2 M/S Cyclic
0x0050	NO	G2 M/S COS /Ack	N/A
0x0051	G2 M/S or G3 Dynamic[1]	G2 M/S COS /Ack	N/A
0x0052	NO	G2 M/S Poll	G2 M/S COS /Ack
0x0053	G2 M/S or G3 Dynamic[1]	G2 M/S Poll	G2 M/S COS /Ack
0x0054	NO	G2 M/S Strobe	G2 M/S COS /Ack
0x0055	G2 M/S or G3 Dynamic[1]	G2 M/S Strobe	G2 M/S COS/Ack

0x0060	NO	G2 M/S Cyclic/Ack	N/A
0x0061	G2 M/S or G3 Dynamic[1]	G2 M/S Cyclic/Ack	N/A
0x0062	NO	G2 M/S Poll	G2 M/S Cyclic/Ack
0x0063	G2 M/S or G3 Dynamic[1]	G2 M/S Poll	G2 M/S Cyclic/Ack
0x0064	NO	G2 M/S Strobe	G2 M/S Cyclic/Ack
0x0065	G2 M/S or G3 Dynamic[1]	G2 M/S Strobe	G2 M/S Cyclic/Ack

- 在状态改变和周期 I/O 连接中，有“ACK”表示有应答，无“ACK”表示无应答
- NO - 不支持，N/A - 表示不适用

1 支持两种显式连接的一种

2 只支持组 3 动态显式信息连接

4.3.5 客户机配置

DNSCAN 可以生成多个对应于多个设备的客户机连接映射，包括看门狗定时器的映射。以下段落描述实现此功能的参数配置，这些参数对应多个客户机连接。

1. 设备标识

在设备列表中包含设备的标识信息。这些参数是可选的，并且每组参数将被独立于具体的设备应用。

表格 4-30 设备标识信息格式

Item	Description
MacId	设备的 MAC ID (0-63)
VendorId	设备的版本 ID
DeviceType	设备类型
ProductCode	设备的产品代码

如果这些参数的任意一个不为 0，DNSCAN 将拿这些参数和对应设备的实际属性进行校验，任何的不匹配将导致 DNSCAN 无法连接到该设备。

如果这些参数为 0，DNSCAN 将从对应的设备中读出属性参数，并相应地修改设备列表中的这些参数，这些参数将被保存起来供以后使用。

2. 客户机连接配置

设备列表中包含客户机连接的配置参数，这些参数规定了连接和设备的通信类型，并按要求配置连接的参数。注意：多个客户机连接分别对应多个设备。

表格 4-31 客户机连接配置参数

Item	Description
ConnectionFlags	连接组合标志字，指定为每个设备打开的连接的类型，每个设备最多可打开 2 个 I/O 连接和 1 个显式信息连接
ExplicitSize	显式信息缓冲长度，如果不使用显式连接就忽略该参数
Output1Size	第一个 I/O 连接的生产连接长度
Input1Size	第一个 I/O 连接的消费连接长度
Output2Size	第二个 I/O 连接的生产连接长度
Input2Size	第二个 I/O 连接的消费连接长度

3. ConnectionFlags

此参数指定了应用所支持的连接类型。

表格 4-32 客户机连接类型参数位定义

Offset	Bit							
	7	6	5	4	3	2	1	0
0	Res.	AKS	CYC	COS	Res.	ST	P	EX
1	Reserved							G3

阴影部分为以后保留

仅限组 2 I/O 连接标志

AKS 应答抑制位

CYC 循环I/O连接

COS 状态改变I/O连接

ST 位选通I/O连接

P 轮询I/O连接

显式信息连接标志

EX 显式信息连接

G3 仅限于组 3 动态显式信息连接（仅用信息组 3 来配置连接 ID）

4. 所支持的连接标志组合

下表列出了所有的连接组合类型。

表格 4-33 客户机连接标志组合列表

Flags	Explicit	I/O1	I/O2
0x0001	G3 Dynamic [2]	N/A	N/A
0x0002	NO	G2 M/S Poll	N/A
0x0003	G2 M/S or G3 Dynamic[1]	G2 M/S Poll	N/A
0x0004	NO	G2 M/S Strobe	N/A
0x0005	G2 M/S or G3 Dynamic[1]	G2 M/S Strobe	N/A
0x0006	NO	G2 M/S Poll	G2 M/S Strobe
0x0007	G2 M/S or G3 Dynamic[1]	G2 M/S Poll	G2 M/S Strobe
0x0010	NO	G2 M/S COS	N/A
0x0011	G2 M/S or G3 Dynamic[1]	G2 M/S COS	N/A
0x0012	NO	G2 M/S Poll	G2 M/S COS
0x0013	G2 M/S or G3 Dynamic[1]	G2 M/S Poll	G2 M/S COS
0x0014	NO	G2 M/S Strobe	G2 M/S COS
0x0015	G2 M/S or G3 Dynamic[1]	G2 M/S Strobe	G2 M/S COS
0x0020	NO	G2 M/S Cyclic	N/A
0x0021	G2 M/S or G3 Dynamic[1]	G2 M/S Cyclic	N/A

0x0022	NO	G2 M/S Poll	G2 M/S Cyclic
0x0023	G2 M/S or G3 Dynamic[1]	G2 M/S Poll	G2 M/S Cyclic
0x0024	NO	G2 M/S Strobe	G2 M/S Cyclic
0x0025	G2 M/S or G3 Dynamic[1]	G2 M/S Strobe	G2 M/S Cyclic
0x0050	NO	G2 M/S COS /Ack	N/A
0x0051	G2 M/S or G3 Dynamic[1]	G2 M/S COS /Ack	N/A
0x0052	NO	G2 M/S Poll	G2 M/S COS /Ack
0x0053	G2 M/S or G3 Dynamic[1]	G2 M/S Poll	G2 M/S COS /Ack
0x0054	NO	G2 M/S Strobe	G2 M/S COS /Ack
0x0055	G2 M/S or G3 Dynamic[1]	G2 M/S Strobe	G2 M/S COS/Ack
0x0060	NO	G2 M/S Cyclic/Ack	N/A
0x0061	G2 M/S or G3 Dynamic[1]	G2 M/S Cyclic/Ack	N/A
0x0062	NO	G2 M/S Poll	G2 M/S Cyclic/Ack
0x0063	G2 M/S or G3 Dynamic[1]	G2 M/S Poll	G2 M/S Cyclic/Ack
0x0064	NO	G2 M/S Strobe	G2 M/S Cyclic/Ack
0x0065	G2 M/S or G3 Dynamic[1]	G2 M/S Strobe	G2 M/S Cyclic/Ack
0x0100	G3 Dynamic[3]	N/A	N/A

- 在状态改变和周期 I/O 连接中，有“ACK”表示有应答，无“ACK”表示无应答
- NO - 不支持，N/A - 表示不适用

1 支持两种显式连接中的一种
2 配置显式信息连接，组 3 动态或预定义的
3 只配置组 3 动态显式信息连接
4 阴影部分保留（以后再扩展这些功能模块）

4.3.6 主站配置

DNSCAN 可以充当组 2 主从/连接组中主站的角色，以下描述的是实现主站功能的一些配置参数。

表格 4-34 主站配置参数

Item	Description
ScanInterval	定期扫描时间参数（通过函数 DNS_Online 设置，单位为毫秒），如果该参数为 0，主站将根据网络配置参数尽可能快的扫描，否则按给定的时间间隔进行扫描。在选通和轮循 I/O 中，当 I/O interval 等于 0 时，EPR 和扫描间隔等于该参数所示的时间值，否则 EPR 和扫描间隔等于 I/O interval，与这里的参数无关。
ReconnectTime	指定 DNSCAN 对出现通信错误的设备进行重连操作的时间间隔，每次重连操作的时间间隔为 ReconnectTime 毫秒，0 表示缺省值 10 秒，有效范围在 100－65535 毫秒之间。

4.3.7 扫描器配置

DNSCAN 是 DeviceNet 网络中的一个物理节点，下面参数的配置是 DNSCAN 能作为多重角色使用的前题条件。

表格 4-35 扫描器配置参数

Item	Description
MAC ID	DNSCAN 的 MAC ID
Baud Rate	DeviceNet 要求的通信波特率

状态指示 LED 反映 DNSCAN 在 DeviceNet 网络中的通信状态。

表格 4-36 扫描器的 LED 状态

COMM Led	Status
全灭	没有打开板卡
绿灯闪烁	在线，但无任何连接被激活
绿灯常亮	在线，至少有一个连接被激活
红灯常亮	没有上线

4.3.8 主站操作

本节描述组 2 主/从 I/O 连接的情况。

“组 2 主/从 IO 扫描时间”主要受 DNSCAN 和以下几个因素影响。

- 被扫描的设备数量
- 网络波特率
- 网络上的 I/O 数据拥挤情况
- 设备超时
- 对仅限组 2 从站的 UCMM 代理情况
- 显式信息的交换量
- 扫描间隔参数

下面描述两种不易理解的情况。

设备超时

如果设备的应答信息不能被较快的接收，那么扫描时间将被延长直到设定的最大值，如果在连续三次扫描周期内都无法接收设备的应答信息，那么该设备被标记为不存在，这样将导致重连动作的发生。

对仅限组2从站的UCMM代理情况

仅限组 2 从站是一种能力有限的设备，它的动态显式信息通信（另一端是非主站设备，如配置工具）由 DNSCAN 代理，当网络上出现大量的显式信息代理通信时，I/O 扫描时间自然就被延长了，比如多个仅限组 2 设备在上传和下载配置参数时。

4.3.9 客户机操作

本节涉及所有的客户机连接。

1. 初始化

当使用函数 DNS_StartScan 启动 DNSCAN 时，相对于每个设备的客户机连接就被生成了，当然必须预先指定连接组合的类型。

2. 设备初始化

DNSCAN 启动后，扫描列表中的每个设备都将顺序地被初始化。

- 打开显式信息连接
- 读取并校验设备标识信息
- 生成 I/O 连接
- 读取连接所能处理的数据长度，存起来备用

3. 设备超时和在线插入

当扫描列表中的设备停止应答或超时时，DNSCAN 可以自动按用户定义的时间间隔进行重连操作，该操作实行和设备初始化一样的动作（一般要先关闭设备的相关连接）。设备的拆除和重新接入操作可以在不中断其和其他设备通信的情况下进行。

4. 客户机显式信息

DNSCAN 可以发送显式请求信息给一个设备，显式请求和应答信息都是通过设备列表中分配给该设备的显式信息缓冲区进行交换。

客户机显式请求信息格式（内部使用）

客户机使用下表显示的数据格式发送显式请求信息给设备列表中的某个设备。

表格 4-37　客户机显式请求信息格式

Name	Data Type	Description
Size	Word	服务参数（Service Data）的数据长度（Byte）
Service	Word	DeviceNet的公共服务代码
Class [1]	Word	所要求的对象分类代码
Instance [1]	Word	所要求对象分类的实例号
Service Data	-	请求服务中可选的服务参数

客户机显式请求信息举例（内部使用）

这是一个关于服务Get_Attribute_Single的示例，如下表所示。

表格 4-38　客户机显式请求信息举例性

Name	Data Type	Value	Description
Size	Word	0x0001	Service Data的数据长度，为1Byte
Service	Word	0x000E	Get_Attribute_Single的服务代码
Class	Word	0x0001	表示标识对象（Identity）
Instance	Word	0x0001	实例号为0x01
Service Data	-	0x07	服务参数0x07表示要求返回产品名称（Product Name）

客户机显式应答信息格式（内部使用）

DNSCAN 使用下表显示的数据格式接收显式应答信息。

表格 4-39 客户机显式应答信息格式

Name	Data Type	Description
Size	Word	Service Data的数据长度（Byte）
Service/Result [1]	Word	DeviceNet服务代码 / 内部错误码
Service Data	-	服务返回的可选应答参数
1 Result codes > 0FFh 被保留做DNSCAN的内部错误码		

客户机显式应答信息举例（内部使用）

这是一个关于服务 Get_Attribute_Single 的成功应答示例，如下表所示。

表格 4-40 客户机显式应答信息举例

Name	Data Type	Value	Description
Size	Word	0x0001	Service Data的数据长度为5 Bytes
Service	Word	0x008E	关于Get_Attribute_Single的应答
Service Data	-	”HELLO”	产品名称

内部错误码（内部使用）

应答域“Service/Result“中的数据大于 0xff 的表示 DNSCAN 的内部错误码，其定义如下表所示。

表格 4-41 内部错误码

Code	Description
0100h	保留
0101h	显式信息体格式不能正确描述所要求的类 (比如类代码 ＞255，而信息体格式为8/8 或 8/16)
0102h	显式信息体格式不能正确描述所要求的实例 (比如实例号 ＞255，而信息体格式为8/8或16/8)
0103h~ffffh	保留

4.3.10 服务器操作

本节描述服务器所有的连接情况。

1. 初始化

服务器连接必须通过外部的客户机设备使用分配预定义组 2 主/从连接组命令或 UCMM 建立。

2. 服务器显式信息

DNSCAN 可以把外面主站的显式请求信息传送给主机中的服务器模块。在服务器模块中，对于指向连接类和连接类实例、DeviceNet 类和 DeviceNet 类实例、Identity 类实例 1 的显式信息将直接被 DNSCAN 处理（不上传到主机应用程序），其他的显式信息将被上传到主机。

如果主机没有配置 DNSCAN 去支持服务器功能或该功能未被使能，DNSCAN 收到任何显式请求信息后都将直接返回一个出错应答，否则任何收到的显式请求信息都将被受理，主机应用程序应及时处理这些显式请求信息并返回一个应答或错误码。

服务器显式请求信息格式（内部使用）

DNSCAN 使用表格 4-42 所示的数据格式传输外来的显式请求信息给主机服务器模块，因为可能有多个请求信息需要被主机应用程序处理，所以需要连接 ID 把相应的显式应答信息和请求信息匹配起来。

表格 4-42 服务器显式请求信息格式

Name	Data Type	Description
ConnectionId	Word	连接 ID，使相应的请求和应答信息匹配
Size	Word	Service Data 包含的数据量（Byte）
Service	Word	DeviceNet 的公共服务代码
Class	Word	类 ID
Instance	Word	实例号
Service Data	-	可选的服务请求参数

服务器显式请求信息举例（内部使用）

这是一个关于服务 Get_Attribute_Single 的示例，如下表所示。

表格 4-43 服务器显式请求信息举例

Name	Data Type	Value	Description
ConnectionId	Word	0x	一个使用信息组 3 的显式请求信息连接 ID
Size	Word	0x0001	Service Data 所含数据量，为 1Byte
Service	Word	0x000E	表示 Get_Attribute_Single 的服务码
Class	Word	0x0064	供应商特定类 ID
Instance	Word	0x0001	实例号为 0x01
Service Data	-	0x01	属性数据为 0x01

服务器显式应答信息格式（内部使用）

主机使用表格 4-44 所示的数据格式来传输显式应答信息，因为可能有多个请求信息需要被主机应用程序处理，所以需要连接 ID 把相应的显式应答信息和请求信息匹配起来。

表格 4-44 服务器显式应答信息格式

Name	Data Type	Description
ConnectionId	Word	连接 ID，使相应的请求和应答信息匹配
Size	Word	Service Data 包含的数据量（Byte）
Service	Word	DeviceNet 的公共服务代码
Service Data	-	可选的服务应答参数

服务器显式应答信息举例（内部使用）

这是一个关于服务 Get_Attribute_Single 的成功应答示例，如下表所示。

表格 4-45 服务器显式应答信息举例

Name	Data Type	Value	Description
ConnectionId	Word	0x	相对于某个请求的连接 ID
Size	Word	0x0002	Service Data 所含数据量，为 2Byte
Service	Word	0x008E	服务 Get_Attribute_Single 的应答
Service Data	-	0xAA55	属性数据

4.3.11 输入/输出环形队列

输入环形队列用于从 DNSCAN 把通信数据传输给主机，输出环形队列用于从主机把通信数据传输给 DNSCAN。该队列存在于主机和 DNSCAN 共享的双端口 RAM 中，由系统自行管理。

4.4 DeviceNet 扫描模块 32-BIT DLL

4.4.1 简介

ZLG_DNSCAN32.DLL 为 DeviceNet 应用程序提供了一个访问和控制 DNSCAN 的通用接口，该接口提供 DeviceNet 通讯服务，但隐藏了模块内部的软硬件实现细节。此 DLL 是一个 32 位的动态链接库，支持 WIN98/Me、NT4、WIN2000/XP 等操作系统，支持在 VC++、C++Builder、Delphi、VB 下进行二次开发。

4.4.2 接口提供的服务

ZLG_DNSCAN32.DLL 提供下列服务：

- DLL 版本信息
- 卡的管理（最多支持 4 块卡、4 个客户机）
- 应用程序无需知道卡的内部资源细节
- 内部集成事件通知
- 使用互锁装置保护共享的 I/O 信息数据
- 使用互锁装置保护共享的显式信息数据

4.4.3 应用分层

应用程序
可选的应用层 API
ZLG_DNSCAN32.DLL
硬件驱动 DLL
WIN32 模式驱动程序（WDM）
硬件

4.5 DeviceNet 主站模块动态链接库接口

4.5.1 简介

下面段落描述了ZLG_DNSCAN32.DLL中的API（应用程序接口）和重要的数据类型定义。应用流程如图 4-1所示。

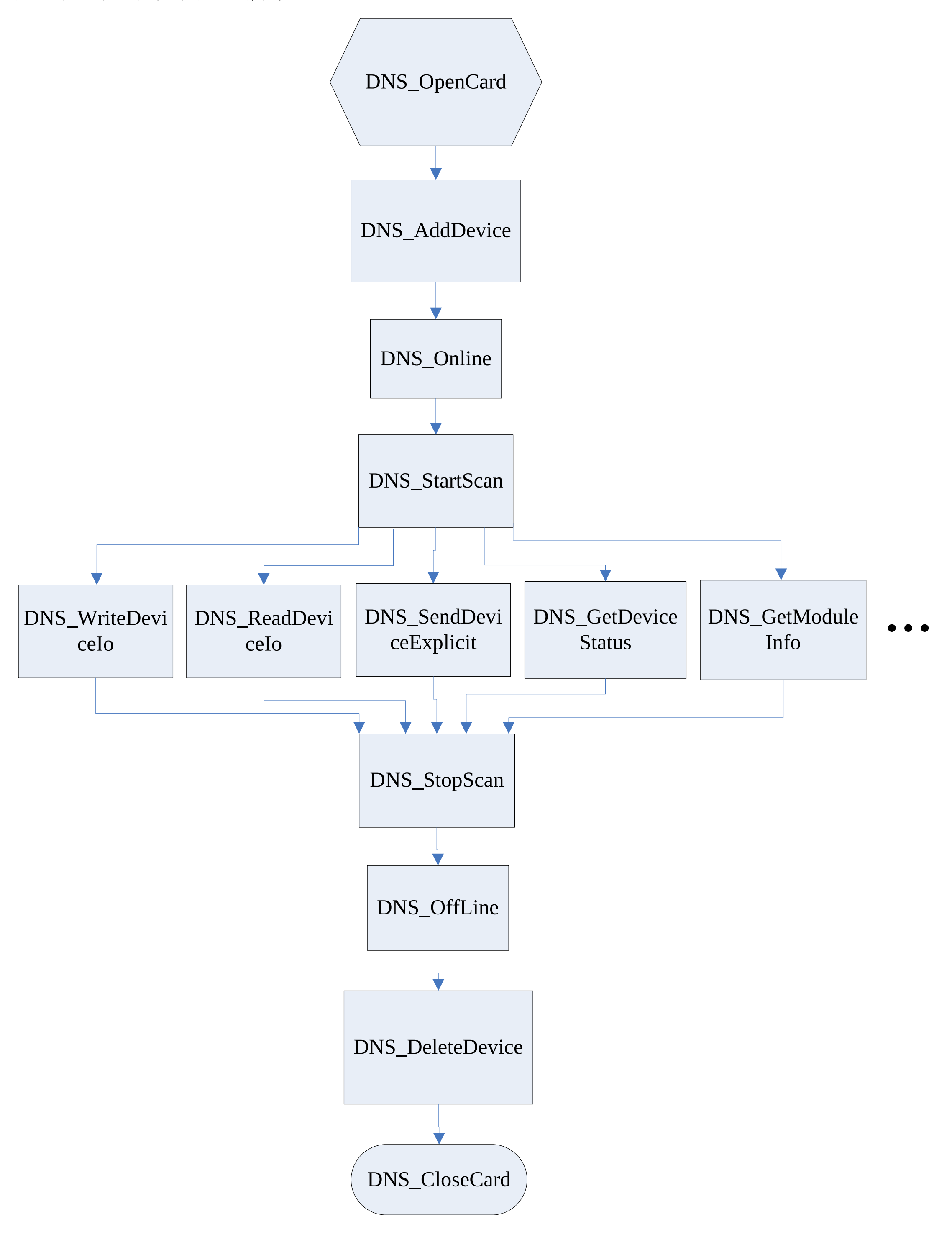


图 4-1 ZLG_DNSCAN32 API 的应用流程

4.5.2 DNS_AddDevice

功能描述：

添加设备到扫描列表。

声明：

BOOL __stdcall DNS_AddDevice(DWORD CardType,DWORD CardInd, DNS_DEVICE_CFG *DeviceCfg)

参数：

Argument	Description
CardType	板卡的类型，固定为0
CardInd	标识同一台机器上不同板卡的序号
DeviceCfg	设备配置块，参考4.5.6 DNS_DEVICE_CFG Data Type

返回值：

Value	Description
TRUE	Success
FALSE	Error，使用 GetLastError 获取错误码

错误码：

Value	Description
2000 0002$_{hex}$	执行命令超时
2000 0003$_{hex}$	没有打开板卡
2000 0007$_{hex}$	空指针
2000 0102$_{hex}$	板卡类型有误或序号冲突
2000 0202$_{hex}$	无效的MAC ID
2000 0205$_{hex}$	相关设备已经存在
2000 020f$_{hex}$	未知错误
2000 021a$_{hex}$	不被支持的连接类型
2000 021b$_{hex}$	无效的连接标志
2000 0220$_{hex}$	无效的信息长度
2000 0221$_{hex}$	无效的选通信息长度

4.5.3 DNS_CAN_COUNTERS Data Type

声明：

```
typedef struct
{
        WORD CanTx;             // CAN transmit error counter
        WORD CanRx;             // CAN recieve error counter
        WORD CanError;          // CAN error Counter
        WORD CanLost;           // CAN lost counter
} DNS_CAN_COUNTERS;
```

4.5.4 DNS_CloseCard

功能描述：

关闭并禁用指定板卡。

声明：

BOOL __stdcall DNS_CloseCard(DWORD CardType,DWORD CardInd)

参数表：

Argument	Description
CardType	板卡的类型，固定为 0
CardInd	标识同一台机器上不同板卡的序号

返回值：

Value	Description
TRUE	Success
FALSE	Error，使用 GetLastError 获取错误码

错误码：

Value	Description
2000 0003_{hex}	没有打开板卡
2000 0102_{hex}	板卡类型有误或序号冲突

4.5.5 DNS_DeleteDevice

功能描述：

删除扫描列表中指定设备，并且关闭和设备相关的所有连接。

声明：

BOOL __stdcall DNS_DeleteDevice(DWORD CardType,DWORD CardInd, WORD DeviceId)

参数表：

Argument	Description
CardType	板卡的类型，固定为 0
CardInd	标识同一台机器上不同板卡的序号
DeviceId	被删除设备的 MAC ID

返回值：

Value	Description
TRUE	Success
FALSE	Error，使用 GetLastError 获取错误码

错误码：

Value	Description
2000 0003_{hex}	没有打开板卡
2000 0102_{hex}	板卡类型有误或序号冲突
2000 0206_{hex}	指定设备不存在
2000 $020f_{hex}$	未知错误
2000 0211_{hex}	设备没有被配置
2000 $021f_{hex}$	无效的设备ID

4.5.6 DNS_DEVICE_CFG Data Type

声明：

```
typedef struct
{
        WORD MacId;                         // Device MacId
        WORD VendorId;                      // Device Vendor Id
        WORD DeviceType;                    // Device Device Type
        WORD ProductCode;                   // Device Product Code
        WORD ProductionInhibitTime;         // Production Inhibit Time for use with
                                            // COS I/O connections   1
        WORD Flags;                         // Connection flags
        WORD Io1Interval;                   // I/O 1 connection interval
        WORD Io2Interval;                   // I/O 2 connection interval

        WORD ExplicitSize;
        WORD Output1Size;
        WORD Input1Size;
        WORD Output2Size;
        WORD Input2Size;

        DWORD Reserved1_timer;              // Reserved for system use
        DWORD Reserved2_timer;              // Reserved for system use
        DWORD Reserved3_timer;              // Reserved for system use
        DWORD Reserved4_timer;              // Reserved for system use
        DWORD Reserved5_timer;              // Reserved for system use
        DWORD Reserved6_timer;              // Reserved for system use

        WORD Reserved1;                     // Reserved for system use
        WORD Reserved2;                     // Reserved for system use
        WORD Reserved3;                     // Reserved for system use
        WORD Reserved4;                     // Reserved
        WORD Reserved1_exp;                 // Reserved for system use
        WORD Reserved2_exp;                 // Reserved for system use
        WORD Reserved3_exp;                 // Reserved for system use
        WORD Reserved4_exp;                 // Reserved for system use
        WORD Reserved1_IO1;                 // Reserved for system use
        WORD Reserved2_IO1;                 // Reserved for system use
        WORD Reserved3_IO1;                 // Reserved for system use
        BYTE *pSend_IO1;                    // Reserved for system use

        WORD Reserved1_IO2;                 // Reserved for system use
        WORD Reserved2_IO2;                 // Reserved for system use
        WORD Reserved3_IO2;                 // Reserved for system use
```

```
        BYTE *pSend_IO2;                    // Reserved for system use

        WORD Reserved1_IO3;                 // Reserved for system use
        WORD Reserved2_IO3;                 // Reserved for system use
        WORD Reserved3_IO3;                 // Reserved for system use
        BYTE *pSend_IO3;                    // Reserved for system use
} DNS_DEVICE_CFG;
```

该参数只对状态改变 I/O 连接有效。

4.5.7 DNS_Driver

功能描述：

读取硬件驱动的版本信息，该信息包括代表版本的数字和描述字符串。

声明：

BOOL __stdcall DNS_Driver(DWORD CardType,DWORD CardInd, TCHAR *Buffer, WORD *Version, DWORD Size)

参数表：

Argument	Description
CardType	板卡的类型，固定为 0
CardInd	标识同一台机器上不同板卡的序号
Buffer	字符串缓冲区指针
Version	低位表示次版本号；高位表示主版本号。
Size	字符串缓冲区长度，若字符串太长则被削去后面部分

返回值：

Value	Description
TRUE	Success
FALSE	Error，使用 GetLastError 获取错误码

错误码：

Value	Description
2000 0003_{hex}	没有打开板卡
2000 0007_{hex}	空指针
2000 0102_{hex}	板卡类型有误或序号冲突

4.5.8 DNS_GetBusStatus

功能描述：

获取 CAN 总线状态字。CAN 总线状态字的相关位信息列表如下：

状态字	Bit							
	7	6	5	4	3	2	1	0
低字节	Reserved	RO	Reserved	Reserved	Reserved	BO	Reserved	Reserved
高字节	SA	O5	O2	O1	Reserved		BP	ER

应用特定标志位

SA – DNSCAN已经被激活（至少有一个I/O连接被建立了）

O5 – 在线且波特率为500 K Bps

O2 -在线且波特率为250 K Bps

O1 -在线且波特率为125 K Bps

公共标志位

BP – 总线供电情况（0表示没有总线供电）

ER – 只要有CAN的通信错误都会置位

RO – 接收缓冲溢出（意味着主机来不及处理低层传来的数据）

BO – 指示因为产生严重错误导致CAN总线掉线

声明：

BOOL __stdcall DNS_GetBusStatus(DWORD CardType,DWORD CardInd,WORD*BusStatus)

参数表：

Argument	Description
CardType	板卡的类型，固定为 0
CardInd	标识同一台机器上不同板卡的序号
BusStatus	指向接收总线状态字的变量地址

返回值：

Value	Description
TRUE	Success
FALSE	Error，使用 GetLastError 获取错误码

错误码：

Value	Description
2000 0003_{hex}	没有打开板卡
2000 0007_{hex}	空指针
2000 0102_{hex}	板卡类型有误或序号冲突

4.5.9 DNS_GetCANCounters

功能描述：

获取 CAN 的计数寄存器。

声明：

BOOL __stdcall DNS_GetCANCounters(DWORD CardType, DWORD CardInd,
DNS_CAN_COUNTERS *CANCounters)

参数表：

Argument	Description
CardType	板卡的类型，固定为 0
CardInd	标识同一台机器上不同板卡的序号
CANCounters	指向接收计数器的缓冲区，参考 0 DNS_CAN_COUNTERS Data Type

返回值：

Value	Description
TRUE	Success
FALSE	Error，使用 GetLastError 获取错误码

错误码：

Value	Description
2000 0003_{hex}	没有打开板卡
2000 0007_{hex}	空指针
2000 0102_{hex}	板卡类型有误或序号冲突

4.5.10 DNS_GetDevice

功能描述：

从扫描列表中读取设备配置信息。

声明：

BOOL __stdcall DNS_GetDevice(DWORD CardType,DWORD CardInd, WORD DeviceId, DNS_DEVICE_CFG *DeviceCfg)

参数表：

Argument	Description
CardType	板卡的类型，固定为 0
CardInd	标识同一台机器上不同板卡的序号
DeviceId	指定被获取配置信息的设备的 MAC ID
DeviceCfg	指向包含设备配置块的缓冲区地址，参考 4.5.6 DNS_DEVICE_CFG Data Type

返回值：

Value	Description
TRUE	Success
FALSE	Error，使用 GetLastError 获取错误码

错误码：

Value	Description
2000 0003_{hex}	没有打开板卡
2000 0007_{hex}	空指针
2000 0102_{hex}	板卡类型有误或序号冲突
2000 0206_{hex}	设备不存在
2000 $020f_{hex}$	未知错误
2000 0211_{hex}	设备没有被配置

2000 021f$_{hex}$	无效的设备ID

4.5.11 DNS_GetDeviceStatus

功能描述：

获取指定设备的状态信息。状态码描述：

Status	Meaning
00h	设备不在扫描列表中
01h	设备空闲(not being scanned)
02h	设备正在被扫描
03h	设备超时
04h	UCMM连接错误
05h	主/从连接组正在忙
06h	分配主/从连接组时出错
07h	无效的供应商ID
08h	读取供应商ID时出错
09h	无效的设备类型
0Ah	读取设备类型时出错
0Bh	无效的产品代码
0Ch	读取产品代码时出错
0Eh	读取I/O连接1的输入长度时出错
10h	读取I/O连接1输出长度时出错
12h	读取I/O连接2的输入长度时出错
14h	读取I/O连接2输出长度时出错
15h	设置I/O连接1expected_packet_rate属性时出错
16h	设置I/O连接2expected_packet_rate属性时出错
18h	设置生产禁止时间时出错
19h	I/O数据溢出
1ah – FFh	保留

声明：

BOOL __stdcall DNS_GetDeviceStatus(DWORD CardType,DWORD CardInd,
WORD DeviceId, BYTE *DeviceStatus)

参数表：

Argument	Description
CardType	板卡的类型，固定为 0
CardInd	标识同一台机器上不同板卡的序号
DeviceId	指定被获取状态信息的设备的 MAC ID
DeviceStatus	指向包含状态信息的缓冲区的指针

返回值：

Value	Description
TRUE	Success
FALSE	Error，使用 GetLastError 获取错误码

错误码：

Value	Description
2000 0003_{hex}	没有打开板卡
2000 0007_{hex}	空指针
2000 0102_{hex}	板卡类型有误或序号冲突
2000 0211_{hex}	设备没有被配置
2000 $021f_{hex}$	无效的设备ID

4.5.12 DNS_GetServerG2Status（保留）

功能描述：

获取组 2 服务器连接组合标志和状态信息。服务器状态码：

Status	Meaning
00h	空闲 (组2主/从连接组未被分配)
01h	激活 (组2主/从连接组已被分配)
02h – FFh	保留

声明：

BOOL __stdcall DNS_GetServerG2Status(DWORD CardType,DWORD CardInd,
BYTE *ServerG2Status, BYTE * ConnectionFlags)

参数表：

Argument	Description
CardType	板卡的类型，固定为 0
CardInd	标识同一台机器上不同板卡的序号
ServerG2Status	指向包含状态信息的缓冲区的指针
ConnectionFlags	指向包含连接组合标志的缓冲区的指针，参考“4.3.4 服务器配置”

返回值：

Value	Description
TRUE	Success
FALSE	Error，使用 GetLastError 获取错误码

错误码：

Value	Description
2000 0003_{hex}	没有打开板卡
2000 0007_{hex}	空指针
2000 0102_{hex}	板卡类型有误或序号冲突
2000 0213_{hex}	服务器没有被配置

4.5.13 DNS_Offline

功能描述：

停止DeviceNet通信并命令DNSCAN下线，在I/O状态为激活时不能执行该函数。

声明：

BOOL __stdcall DNS_Offline(DWORD CardType,DWORD CardInd)

参数表：

Argument	Description
CardType	板卡的类型，固定为 0
CardInd	标识同一台机器上不同板卡的序号

返回值：

Value	Description
TRUE	Success
FALSE	Error，使用 GetLastError 获取错误码

错误码：

Value	Description
2000 0003_{hex}	没有打开板卡
2000 0102_{hex}	板卡类型有误或序号冲突
2000 0208_{hex}	在I/O激活状态下该命令不能被执行
2000 $020f_{hex}$	未知错误

4.5.14 DNSOnline

功能描述：

设置DeviceNet通信和服务参数并命令DNSCAN上线。

声明：

BOOL __stdcall DNS_Online(DWORD CardType,DWORD CardInd, DNS_SCANNER_CFG *ScannerCfg)

参数表：

Argument	Description
CardType	板卡的类型，固定为 0
CardInd	标识同一台机器上不同板卡的序号
ScannerCfg	DNSCAN 的配置数据，参考 0 DNS_SCANNER_CFG Data Type

返回值：

Value	Description
TRUE	Success
FALSE	Error，使用 GetLastError 获取错误码

错误码：

Value	Description
2000 0003_{hex}	没有打开板卡
2000 0007_{hex}	空指针
2000 0102_{hex}	板卡类型有误或序号冲突
2000 0202_{hex}	无效的MAC ID
2000 0203_{hex}	无效的波特率
2000 0204_{hex}	重复的MAC ID
2000 0209_{hex}	已经在线
2000 $020f_{hex}$	未知错误

2000 021b$_{hex}$	无效的连接标志
2000 0220$_{hex}$	无效的信息长度
2000 0226$_{hex}$	无应答的CAN消息

4.5.15 DNS_OpenCard

功能描述：

打开主站卡，在执行所有其它函数之前必须打开主站卡。

声明：

BOOL __stdcall DNS_OpenCard(DWORD CardType,DWORD CardInd)

参数表：

Argument	Description
CardType	板卡的类型，固定为 0
CardInd	标识同一台机器上不同板卡的序号

返回值：

Value	Description
TRUE	Success
FALSE	Error，使用 GetLastError 获取错误码

错误码：

Value	Description
2000 0001$_{hex}$	无法找到合适的驱动
2000 0008$_{hex}$	已经打开板卡

4.5.16 DNS_ReadDeviceIo

功能描述：

读取主站扫描列表中某个设备的I/O数据。

相关标题：

参考 4.3.5 客户机配置。

声明：

BOOL __stdcall DNS_ReadDeviceIo(DWORD CardType,DWORD CardInd, WORD DeviceId, BYTE IoArea, void *Buffer,WORD Size, int WaitTime=1000)

参数表：

Argument	Description
CardType	板卡的类型，固定为 0
CardInd	标识同一台机器上不同板卡的序号
DeviceId	被读取 I/O 数据的设备的 ID
IoArea	指定设备中被读取的 I/O 数据域
Buffer	指向接收数据缓冲区的指针
Size	接收缓冲区的长度（Byte），必须包含精确的长度值
WaitTime	接收缓冲区没有数据时的等待超时(ms)，缺省为 1 秒

参数表中的IoArea：

Value	Name	Description
1	DNS_INPUT1	I/O1的输入数据域1

2	DNS_INPUT2	I/O2的输入数据域1
3-0xff	保留	

返回值：

Value	Description
TRUE	Success
FALSE	Error，使用 GetLastError 获取错误码

错误码：

Value	Description
2000 0002_{hex}	读数据超时
2000 0003_{hex}	没有打开板卡
2000 0007_{hex}	空指针
2000 0102_{hex}	板卡类型有误或序号冲突
2000 0110_{hex}	输入的数据长度无效
2000 0210_{hex}	无效的数据域
2000 0211_{hex}	设备没有被配置
2000 $021f_{hex}$	无效的设备ID
2000 0222_{hex}	连接没有被配置

4.5.17 DNS_ReadServerIo（保留）

功能描述：

读取DNSCAN作为服务器使用时的I/O数据。

相关标题：

参考 4.3.4 服务器配置。

声明：

BOOL __stdcall DNS_ReadServerIo(DWORD CardType,DWORD CardInd, BYTE IoArea, void *Buffer, WORD Size,int WaitTime=1000)

参数表：

Argument	Description
CardType	板卡的类型，固定为 0
CardInd	标识同一台机器上不同板卡的序号
IoArea	指定服务器中被读取的 I/O 数据域
Buffer	指向接收数据缓冲区的指针
Size	接收缓冲区的长度（Byte），必须包含精确的长度值
WaitTime	接收缓冲区没有数据时的等待超时，缺省为 1 秒

参数表中的IoArea：

Value	Name	Description
1	DNS_INPUT1	I/O1的输入数据域1
2	DNS_INPUT2	I/O2的输入数据域1
3-0xff	保留	

返回值：

Value	Description
TRUE	Success
FALSE	Error，使用 GetLastError 获取错误码

错误码：

Value	Description
2000 0003$_{hex}$	没有打开板卡
2000 0007$_{hex}$	空指针
2000 0102$_{hex}$	板卡类型有误或序号冲突
2000 0110$_{hex}$	输入的数据长度无效
2000 0210$_{hex}$	无效的数据域
2000 0213$_{hex}$	服务器没有被配置
2000 0222$_{hex}$	连接没有被配置

4.5.18 DNS_ReceiveServerExplicit（保留）

功能描述：

读取远端主站发来的显式请求信息。

相关标题：

参考 0

DNS_SendServerExplicit（保留） 和 4.3.10 服务器操作。

声明：

BOOL __stdcall DNS_ReceiveServerExplicit(DWORD CardType,DWORD CardInd,
WORD *ConnectionId, BYTE *Service,WORD *ClassId,
WORD *InstanceId, void *ServiceData, WORD *Size ,int WaitTime=1000)

参数表：

Argument	Description
CardType	板卡的类型，固定为 0
CardInd	标识同一台机器上不同板卡的序号
ConnectionId	分配给该显式请求信息的连接 ID
Service	和显式应答信息相关的服务码
ClassId	显式请求信息指向的类 ID
InstanceId	显式请求信息指向的类实例 ID
ServiceData	指向接收数据缓冲区的指针，参考 DeviceNet 规范中特定类或实例相关的显式请求信息的服务数据区
Size	接收缓冲区的长度（Byte）
WaitTime	接收缓冲区没有数据时的等待超时，缺省为 1 秒

返回值：

Value	Description
TRUE	Success
FALSE	Error，使用 GetLastError 获取错误码

错误码：

Value	Description

2000 0003_{hex}	没有打开板卡
2000 0007_{hex}	空指针
2000 0102_{hex}	板卡类型有误或序号冲突
2000 0213_{hex}	服务器没有被配置
2000 0215_{hex}	连接不存在
2000 0216_{hex}	无效的分类
2000 0217_{hex}	无效的分类实例
2000 0218_{hex}	显式信息无效
2000 0222_{hex}	连接没有被配置

4.5.19 DNS_SCANNER_CFG Data Type

声明：

```
typedef struct
{
        WORD MacId;                     // Client/Server MacId
        WORD BaudRate;                  // Client/Server baud rate
        WORD ScanInterval;              // Global Scan interval
        WORD ReconnectTime              //Reconnnection Timeout Interval
        WORD Flags;                     // Connection flags

        WORD ExplicitRequestSize;
        WORD ExplicitResponseSize;
        WORD Output1Size;
        WORD Input1Size;
        WORD Output2Size;
        WORD Input2Size;
        WORD Reserved;                  // Reserved for system use
        WORD Reserved1_exp;             // Reserved for system use
        WORD Reserved2_exp;             // Reserved for system use
        WORD Reserved3_exp;             // Reserved for system use
        WORD Reserved4_exp;             // Reserved for system use

        WORD Reserved1_IO1;             // Reserved for system use
        WORD Reserved2_IO1;             // Reserved for system use
        WORD Reserved3_IO1;             // Reserved for system use

        WORD Reserved1_IO2;             // Reserved for system use
        WORD Reserved2_IO2;             // Reserved for system use
        WORD Reserved3_IO2;             // Reserved for system use
} DNS_SCANNER_CFG;
```

相关标题：

参考 4.3.4 服务器配置。

4.5.20 DNS_SendDeviceExplicit

功能描述：

给扫描列表中的某个设备发送显式请求信息，同时接收返回的应答信息。

相关标题：

参考4.3.9 客户机操作。

声明：

BOOL __stdcall DNS_SendDeviceExplicit(DWORD CardType,DWORD CardInd,
WORD DeviceId, BYTE Service, WORD ClassId,
WORD InstanceId, void *ServiceData,
WORD Size,void *ReceiveData,
WORD *ReceSize,BYTE *rService,int WaitTime=1000)

参数表：

Argument	Description
CardType	板卡的类型，固定为 0
CardInd	标识同一台机器上不同板卡的序号
DeviceId	接收显式信息的设备的 MAC ID
Service	DeviceNet 服务码
ClassId	分类 ID
InstanceId	分类的实例号
ServiceData	指向服务所需的参数缓冲区的指针，参考 DeviceNet 规范中特定类或实例相关的显式请求信息的服务数据区
Size	缓冲区 ServiceData 的长度
ReceiveData	指向接收数据缓冲区的指针，参考 DeviceNet 规范中特定类或实例相关的显式应答信息的服务数据区
ReceSize	接收缓冲区的长度（Byte），函数返回时表示接收数据缓冲区中的有效数据长度。
rService	实际返回的服务码，如果返回 0x14，那么接收缓冲区包含 DeviceNet 的错误码（参考 DeviceNet 规范卷 I 的附录 H）。
WaitTime	接收缓冲区没有数据时的等待超时(ms)，缺省为 1 秒

返回值：

Value	Description
TRUE	Success
FALSE	Error，使用 GetLastError 获取错误码

错误码：

Value	Description
2000 0003$_{hex}$	没有打开板卡
2000 0007$_{hex}$	空指针
2000 0102$_{hex}$	板卡类型有误或序号冲突
2000 0211$_{hex}$	设备没有被配置
2000 0214$_{hex}$	无效的数据缓冲区长度
2000 0215$_{hex}$	连接不存在

2000 0216_{hex}	无效的分类
2000 0217_{hex}	无效的分类实例
2000 0218_{hex}	显式信息无效
2000 0219_{hex}	显式信息正在被挂起
2000 $021f_{hex}$	无效的设备ID
2000 0222_{hex}	连接没有被配置

4.5.21 DNS_SendServerExplicit（保留）

功能描述：

DNSCAN作为服务器使用时发送显式应答信息。

相关标题：

参考4.3.10 服务器操作。

声明：

BOOL __stdcall DNS_SendServerExplicit(DWORD CardType,DWORD CardInd,
WORD ConnectionId, BYTE Service,
void *ServiceData, WORD Size)

参数表：

Argument	Description
CardType	板卡的类型，固定为 0
CardInd	标识同一台机器上不同板卡的序号
ConnectionId	发送该显式信息的连接的连接 ID
Service	DeviceNet 服务码
ServiceData	指向服务所需的参数缓冲区的指针，参考 DeviceNet 规范中特定类或实例相关的显式应答信息的服务数据区
Size	缓冲区 ServiceData 的长度

返回值：

Value	Description
TRUE	Success
FALSE	Error，使用 GetLastError 获取错误码

错误码：

Value	Description
2000 0003_{hex}	没有打开板卡
2000 0007_{hex}	空指针
2000 0102_{hex}	板卡类型有误或序号冲突
2000 0213_{hex}	服务器没有被配置
2000 0222_{hex}	连接没有被配置

4.5.22 DNS_StartScan

功能描述：

开始执行I/O扫描。

声明：

BOOL __stdcall DNS_StartScan(DWORD CardType,DWORD CardInd)

参数表：

Argument	Description
CardType	板卡的类型，固定为 0
CardInd	标识同一台机器上不同板卡的序号

返回值：

Value	Description
TRUE	Success
FALSE	Error，使用 GetLastError 获取错误码

错误码：

Value	Description
2000 0003_{hex}	没有打开板卡
2000 0102_{hex}	板卡类型有误或序号冲突
2000 0207_{hex}	总线掉线
2000 $020a_{hex}$	DNSCAN已经在运行中
2000 $020c_{hex}$	DNSCAN正在停止
2000 $020f_{hex}$	未知错误

4.5.23 DNS_StopScan

功能描述：

停止I/O扫描。

声明：

BOOL __stdcall DNS_StopScan(DWORD CardType,DWORD CardInd)

参数表：

Argument	Description
CardType	板卡的类型，固定为 0
CardInd	标识同一台机器上不同板卡的序号

返回值：

Value	Description
TRUE	Success
FALSE	Error，使用 GetLastError 获取错误码

错误码：

Value	Description
2000 0003_{hex}	没有打开板卡
2000 0102_{hex}	板卡类型有误或序号冲突
2000 $020b_{hex}$	DNSCAN没有运行
2000 $020c_{hex}$	DNSCAN正在停止
2000 $020f_{hex}$	未知错误

4.5.24 DNS_Version

功能描述：

获取动态链接库的版本信息，该函数将返回一个数字和一个描述字符串。

声明：

BOOL __stdcall DNS_Version(TCHAR *Buffer, WORD *Version, DWORD Size)

参数表：

Argument	Description
Buffer	指向 DLL 信息字符串缓冲区的指针
Version	次版本号在低位；主版本号在高位。
Size	字符串缓冲区长度，若字符串太长则被削去后面部分

返回值：

Value	Description
TRUE	Success
FALSE	Error，使用 GetLastError 获取错误码

错误码：

Value	Description
2000 0007_{hex}	空指针

4.5.25 DNS_WriteDeviceIo

功能描述：

给主站扫描列表中某个设备发送I/O数据。

注意：系统初始化好后，只需调用该函数一次，系统就会自动记录通过该函数传入的发送数据，进而按I/O扫描间隔自动重发，以后只要不断调用DNS_ReadDeviceIo()读取I/O数据。

相关标题：

参考 4.3.5 客户机配置。

声明：

BOOL __stdcall DNS_WriteDeviceIo(DWORD CardType,DWORD CardInd, WORD DeviceId,
BYTE IoArea, void *Data,WORD Size)

参数表：

Argument	Description
CardType	板卡的类型，固定为 0
CardInd	标识同一台机器上不同板卡的序号
DeviceId	被发送 I/O 数据的设备的 MAC ID
IoArea	指定设备中被发送的 I/O 数据域
Data	指向待发送数据缓冲区的指针
Size	发送缓冲区的长度（Byte），必须包含精确的长度值

参数表中的IoArea：

Value	Name	Description
1	DNS_OUTPUT1	I/O1的输出数据域1
2	DNS_OUTPUT2	I/O2的输出数据域1
3-0xff	保留	

返回值：

Value	Description
TRUE	Success
FALSE	Error，使用 GetLastError 获取错误码

错误码：

Value	Description
2000 0003_{hex}	没有打开板卡
2000 0007_{hex}	空指针
2000 0102_{hex}	板卡类型有误或序号冲突
2000 0110_{hex}	输入的数据长度无效
2000 0210_{hex}	无效的数据域
2000 0211_{hex}	设备没有被配置
2000 $021f_{hex}$	无效的设备ID
2000 0222_{hex}	连接没有被配置

4.5.26 DNS_WriteServerIo（保留）

功能描述：

发送DNSCAN作为服务器使用时的I/O数据。

相关标题：

参考 4.3.4 服务器配置。

声明：

BOOL __stdcall DNS_WriteServerIo(DWORD CardType,DWORD CardInd, BYTE IoArea, void *Data, WORD Size)

参数表：

Argument	Description
CardType	板卡的类型，固定为 0
CardInd	标识同一台机器上不同板卡的序号
IoArea	指定服务器中被发送的 I/O 数据域
Data	指向待发送数据缓冲区的指针
Size	发送缓冲区的长度（Byte），必须包含精确的长度值

参数表中的IoArea：

Value	Name	Description
1	DNS_OUTPUT1	I/O1的输出数据域1
2	DNS_OUTPUT2	I/O2的输出数据域1
3-0xff	保留	

返回值：

Value	Description
TRUE	Success
FALSE	Error，使用 GetLastError 获取错误码

错误码：

Value	Description
2000 0003_{hex}	没有打开板卡

2000 0007hex	空指针
2000 0102hex	板卡类型有误或序号冲突
2000 0110hex	输入的数据长度无效
2000 0210hex	无效的数据域
2000 0213hex	服务器没有被配置
2000 0222hex	连接没有被配置

4.5.27 DNS_GetModuleInfo

功能描述：

获取模块公共信息。

数据描述：

Name	Data Type	Description
WinSize	Word	指示共享内存的大小，0=8K,1=16K,2=32K,3=64K
ModuleRev	Word	模块版本，小数点定在两个 Byte 之间，如 0100h 表示 V1.0
CardType	byte[16]	卡的类型，举例：“DN-PCI51XX”
CardSerial	byte[8]	卡的序列号

声明：

BOOL __stdcall DNS_GetModuleInfo(DWORD CardType,DWORD CardInd,
DNS_MODULE_HEADER *ModuleHeader)

参数表：

Argument	Description
CardType	板卡的类型，固定为 0
CardInd	标识同一台机器上不同板卡的序号
ModuleHeader	指向包含应用模块头信息的缓冲区的指针， 参考 5.29 DNS_MODULE_HEADER Data Type

返回值：

Value	Description
TRUE	Success
FALSE	Error，使用 GetLastError 获取错误码

错误码：

Value	Description
2000 0003hex	没有打开板卡
2000 0007hex	空指针
2000 0102hex	板卡类型有误或序号冲突

4.5.28 DNS_MODULE_HEADER Data Type

声明：

```
typedef struct
{
    WORD WinSize;                // Host interface window size
    WORD ModuleRev;              // Module revision
    CHAR CardType[40];           // Card type (i.e. DN-PCI51XX)
    CHAR CardSerial[20];         // Card serial number
} DNS_MODULE_HEADER;
```

5. 测试工具

工具软件“ZLG_Dnsw32.exe”说明了 DNSCAN 动态链接库的接口用法，能够演示并测试主站模块功能。为了使用户能理解 DeviceNet 操作流程，方便开发 DeviceNet 应用系统，在产品光盘中列出了“ZLG_Dnsw32.exe”软件的 C++Builder 5 源码，供用户参考。

以下软件“ZLG_Dnsw32.exe”的操作说明。整个过程按常规的 DeviceNet 使用顺序进行描述。

打开 ZLG_Dnsw32.exe 软件，主界面如图 5-1 所示。如果 DN-PCI-5110 主站卡和驱动程序已经安装好，那么在主界面最下面的状态栏应该显示“Card opened，Offline”。

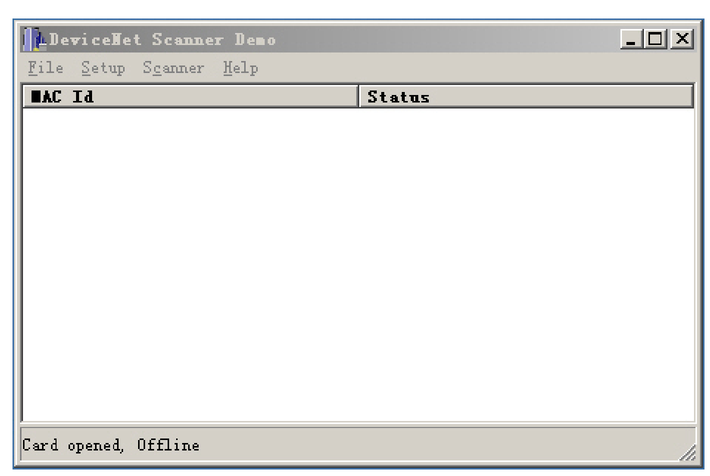


图 5-1　主界面

如图 5-2 所示打开上线窗口，主站在进行网络通信前必须先执行上线操作，而且至少要有一个和主站有相同波特率的节点已经接入网络，该节点不必已经上线。

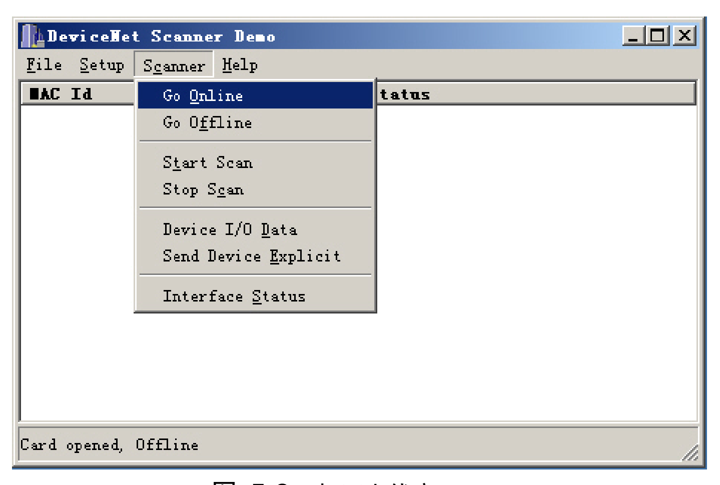


图 5-2　打开上线窗口

上线窗口如图 5-3 所示，

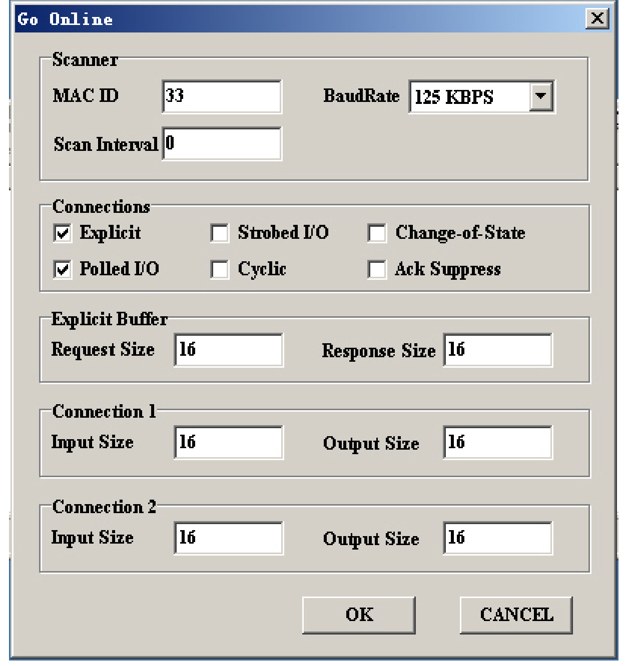


图 5-3　上线窗口

该窗口主要由“Scanner”、“Connections”、“Explicit”、“Connection1”和“Connection2”五个选项组构成。需要注意的是连接组合选项框“Connections”的设置方法，参见 4.3.4 服务器配置。“Connections”的每一个复选框代表着连接组合标志字的一个位（bit），选中表示置 1，否则为 0，这样构成一个具体的标志字，然后通过查找“表格 4-29　服务器连接标志组合列表”所示标志字的值所对应的连接类型，就可以知道主站所支持的连接组合。“Scanner”中编辑框“Scan Interval”的数值所对应的参数和 I/O 扫描间隔参数有一定关系，具体描述请参考 4.3.6 主站配置中“表格 4-34　主站配置参数”。这里我们选择 125 KBps 的波特率，正常上线后，主界面最下面的状态栏应该显示“125K ，Online”。

按图 5-4 所示打开扫描列表编辑窗口。

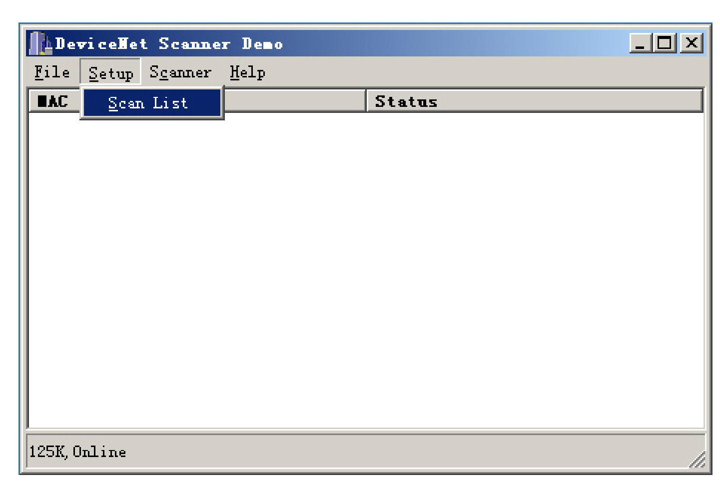


图 5-4　打开扫描列表编辑窗口

刚打开的扫描列表窗口如图 5-5 所示，单击按键“ADD”，可在弹出的子窗口中输入 MAC ID 值，此时按键“DEL”、“PREV”、“NEXT”就可以使用了。

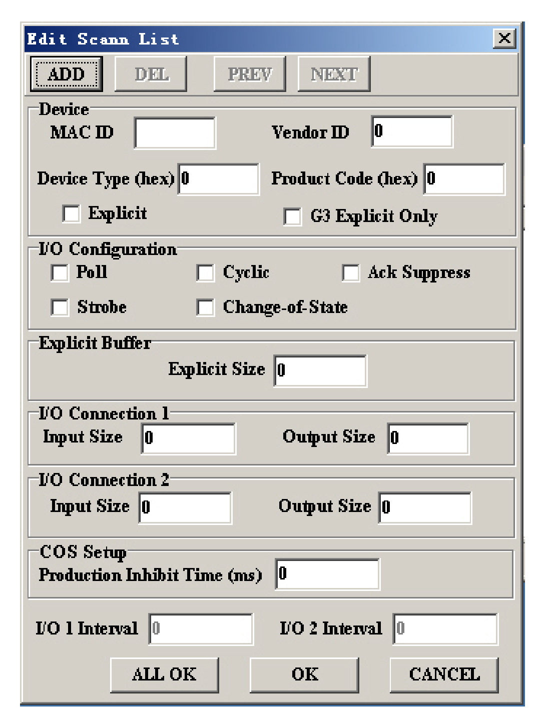


图 5-5 扫描列表窗口

下面介绍图 5-5 中各个选项框的设置方法。

- 选项框“Device”

对于编辑框“Vender ID”、“Device Type”、“Product Code”中被填写为 0 的项，主站在连接设备的过程中将不做和这些项相关的参数校验。如果输入的参数不为 0，那么必须填写正确的参数值，否则将无法正确连接到对应的设备。

- 选项框“I/O Configuration”

这里的复选框和选项框“Device”中的复选框“Explicit”、“G3 Explicit Only”组合起来构成设备连接组合标志字，每一个复选框代表着连接组合标志字的一个位（bit），选中表示置1，否则为0。参考4.3.5 客户机配置，从“表格 4-33 客户机连接标志组合列表”中可以查到该设备所支持的连接组合。

- 选项框“Explicit Buffer”

设置主站对设备列表中的设备进行显式信息收发时所能使用的最大数据量，单位为BYTE。

- 选项框“I/O Connection 1”、“I/O Connection 2”

分别用于设置I/O连接1和I/O连接2的输入输出长度，单位为BYTE。其中“Input”表示从设备到主站，对应设备的生产连接长度；“Output”表示从主站到设备，对应设备的消费连接长度。

- 选项框“COS Setup”

该选项框只对状态变化（change-of-state）有效，用于设置从站连续两次生产数据之间的最小时间间隔。

- 编辑框“I/O 1 Interval”、“I/O 2 Interval”

分被用于设置 I/O 连接 1 和 I/O 连接 2 的 I/O 数据扫描间隔，在 DeviceNet 规范中，这个扫描间隔简称 EPR。

设置好设备参数后，单击按键“OK”就可以把设备添加到扫描列表中，注意，此时设备还没有被扫描。这样连续操作多次，可以把多个设备添加到扫描列表中。这时我们可以把

扫描列表信息用文件存起来，在以后通过打开文件操作就可以把设备添加进来，然后通过按键“ALL OK”一次性把所有设备添加到扫描列表。

设备被添加到扫描列表后，它的 MAC ID 和状态值（Status）将显示在主界面的列表视图中，如图 5-6 所示，选择菜单“Start Scan”执行扫描操作。

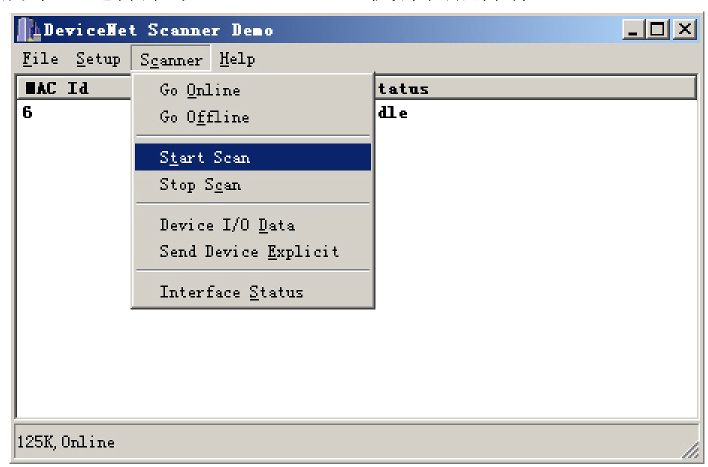


图 5-6 执行扫描操作

经过一定时间后，设备就被激活，如图 5-7 所示。

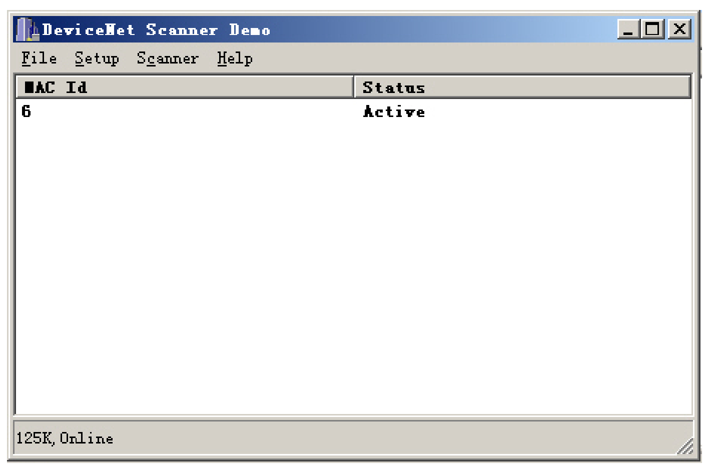


图 5-7 设备正在被扫描

如图 5-8 所示打开 I/O 信息收发窗口，

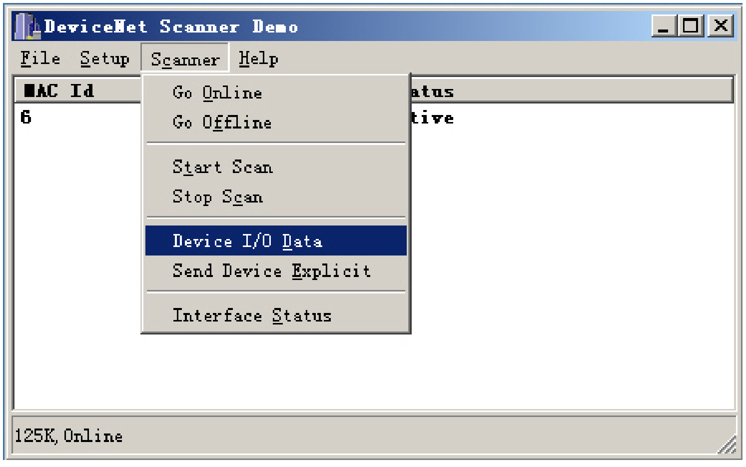


图 5-8 设备 I/O 信息

根据图 5-9所示的I/O信息窗口，我们可以知道，MAC ID为6的设备当前总共有两个I/O连接被激活，I/O连接1的输入输出长度都是1 BYTE，I/O连接2的输入长度是1 BYTE，输出长度是8 BYTE。其中“In”表示从设备到主站，对应设备的生产连接长度；“Out”表示从主站到设备，对应设备的消费连接长度。这两个I/O连接在系统内部被自动维护起来，主站会根据用户输入的I/O扫描间隔定期地给该设备发送I/O信息，所发的I/O信息就是I/O信息窗口上显示的数值，也就是全0。我们可以手工修改I/O信息窗口上的发送数据（对应“Out”部分），从而改变发送到设备的I/O信息。

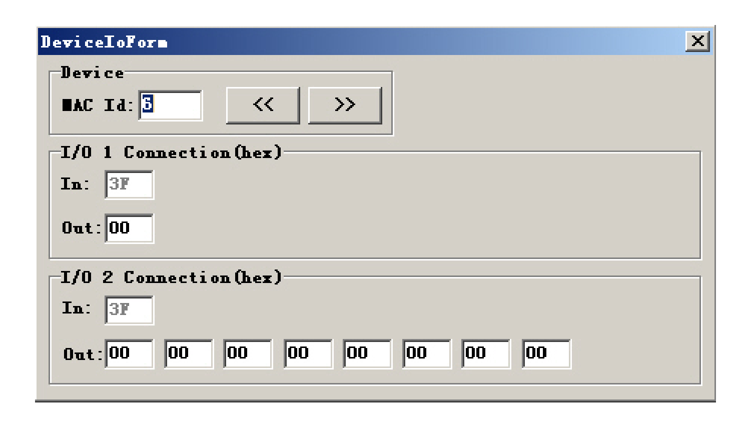


图 5-9 I/O 信息窗口

关闭I/O信息窗口，然后参照附图8所示，选择菜单项“Send Device Explicit”打开显式信息收发窗口，如图 5-10所示。该窗口显示的是获取MAC ID为6的设备的产品名称的处理数据，DeviceNet服务码（Service）为0x0e，表示获取单个属性，类代码（Class Id）为0x01，表示标识符类，实例号（Instance Id）为0x01，可选数据区长度（Data Size）为1 BYTE，数据为0x07，表示产品名称属性ID。输入完发送参数后，点击按键“Send”就可以以编辑框“Count”的值为次数，以编辑框“Interval”的值为每次的发送间隔，不停地发送显式信息，然后接收处理，直到处理完指定次数（Count）。每次成功发送都会有如下所示的显示，“SEND:”后表示发送的数据，“RECE:”后表示接收到的数据，紧跟在“RECE:”后的第一个BYTE表示回应的DeviceNet服务码或错误码，其他数据的格式请参照DeviceNet规范文档。

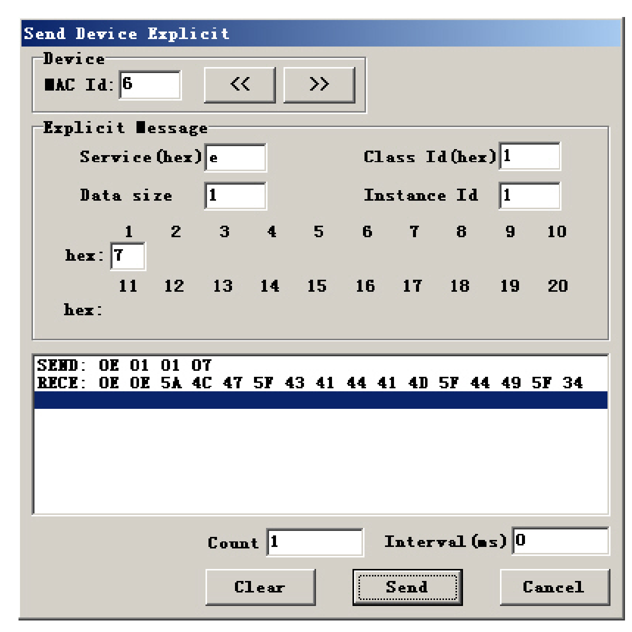


图 5-10 设备显式信息

6. 检查和维护

在 DN-PCI-5110 DeivceNet 主站卡中，主要电气部件都是半导体元件，尽管它有很长的寿命，但在不正确环境下也可能加速老化。应该进行定期检查，以保证保持所要求的条件。推荐每 6 个月到一年，至少检查一次。在不利的环境条件下，应该进行更频繁的检查。

如果在维护过程中遇到问题，请阅读下面的内容，以便找到问题的可能的原因。如果仍无法解决问题，请联系广州致远电子有限公司。

序号	项目	检查	标准	行动
1	电源供应	在电源供应端检查电压波动	电压必须在允许电压波动范围内：+5V DC	使用电压表在电源输入端检查电源。采取必要措施使电压波动在范围之内
3	周围环境	检查周围环境温度（包括封闭环境的内部温度）	0°C ~ +70°C	使用温度计检查温度并确保环境温度保持在允许的范围内
		检查环境湿度（包括封闭环境的内部湿度）	没有空调时相对湿度必须在 10% ～ 90%	使用湿度计检查湿度并确保环境湿度保持在允许范围内
		检查灰尘、粉末、盐、金属屑的积累	没有积累	清洁并保护设备
		检查水、油或化学喷雾碰撞到设备	没有喷雾碰到设备	如果需要清洁保护设备
		检查在设备区域中易腐蚀或易燃气体	没有易腐蚀或易燃气体	通过闻或使用一个传感器检查
		检查震动和冲击水平	震动和冲击在规定范围内	如果需要安装衬垫或其它减震装置
		检查设备附近的噪声源	没有重要噪声信号源	隔离设备和噪声源或保护设备
4	安装和接线	检查每个单元的连接并已经与下一个单元安全锁定	没有松动	把连接器完全压到一起和用滑块把它们锁住
		检查电缆连接器完全插入和锁住	没有松动	纠正任何不正确安装连接器
		检查外部接线中是否有松动螺丝钉	没有松动	用螺丝起子拧紧螺丝钉
		检查外部接线中的压接连接器	在连接器间有足够的空间	肉眼检查如果有必要则调节
		检查外部线电缆的损坏	没有损坏	肉眼检查和如果有必须则替换电缆

7. 免责声明

本文档提供有关致远电子产品的信息。本文档并未授予任何知识产权的许可，并未以明示或暗示，或以禁止发言或其它方式授予任何知识产权许可。除致远电子在其产品的销售条款和条件中声明的责任之外，致远电子概不承担任何其它责任。并且，致远电子对致远电子产品的销售和／或使用不作任何明示或暗示的担保，包括对产品的特定用途适用性、适销性或对任何专利权、版权或其它知识产权的侵权责任等，均不作担保。致远电子产品并非设计用于医疗、救生或维生等用途。致远电子可能随时对产品规格及产品描述做出修改，恕不另行通知。

DN-PCI-5110 DeivceNet 主站卡可能包含某些设计缺陷或错误，一经发现将收入勘误表，并因此可能导致产品与已出版的规格有所差异。如客户索取，可提供最新的勘误表。

在订购产品之前，请您与当地的致远电子销售处或分销商联系，以获取最新的规格说明。本文档中提及的含有订购号的文档以及其它致远电子文献可通过访问广州致远电子有限公司的万维网站点获得，网址是：

http://www.embedcontrol.com/ 或致电+86-20-22644249 查询。

8. 附录 A

表格 8-1　SJA1000 标准波特率

序号	Baudrate (Kbps)	晶振频率=16MHz	
		BTR0 (Hex)	BTR1 (Hex)
1	5	BF	FF
2	10*	31	1C
3	20*	18	1C
4	40	87	FF
5	50*	09	1C
6	80	83	FF
7	100*	04	1C
8	125*	03	1C
9	200	81	FA
10	250*	01	1C
11	400	80	FA
12	500*	00	1C
13	666	80	B6
14	800*	00	16
15	1000*	00	14

注：带*号的是 CIA 协会推荐的波特率。

9. 销售与服务网络

广州致远电子有限公司

地址：广州市天河区车陂路黄洲工业区 3 栋 2 楼　　邮编：510660

电话：(020) 22644249　28872524　22644399　28872342
28872349　28872569　28872573

传真：**(020) 38601859**

网站：**www.embedtools.com**　**www.embedcontrol.com**　**www.ecardsys.com**

ZHIYUAN ELECTRONICS

广州周立功单片机发展有限公司

地址：广州市天河北路 689 号光大银行大厦 15 楼 F1　邮编：510630

电话：(020)38730916　38730917　38730976　38730977

传真：(020)38730925

网址：http://www.zlgmcu.com

ZLG

广州专卖店

地址：广州市天河区新赛格电子城 203-204 室

电话：(020)87578634　87569917

传真：(020)87578842

南京周立功

地址：南京市珠江路 280 号珠江大厦 2006 室

电话：(025)83613221　83613271　83603500

传真：(025)83613271

北京周立功

地址：北京市海淀区知春路 113 号银网中心 712 室（中发电子市场斜对面）

电话：(010)62536178　62536179　82628073

传真：(010)82614433

重庆周立功

地址：重庆市石桥铺科园一路二号大西洋国际大厦（赛格电子市场）1611 室

电话：(023)68796438　68796439

传真：(023)68796439

杭州周立功

地址：杭州市登云路 428 号浙江时代电子市场 205 号

电话：(0571)88009205　88009932　88009933

传真：(0571)88009204

成都周立功

地址：成都市一环路南一段 57 号金城大厦 612 室

电话：(028)85499320　85437446

传真：(028)85439505

深圳周立功

地址：深圳市深南中路 2070 号电子科技大厦 A 座 24 楼 2403 室

电话：(0755)83781768　83781788　83782922

传真：(0755)83793285

武汉周立功

地址：武汉市洪山区广埠屯珞瑜路 158 号 12128 室（华中电脑数码市场）

电话：(027)87168497　87168297　87168397

传真：(027)87163755

上海周立功

地址：上海市北京东路 668 号科技京城东座 7E 室

电话：(021)53083452　53083453　53083496

传真：(021)53083491

西安办事处

地址：西安市长安北路 54 号太平洋大厦 1201 室

电话：(029)87881296　83063000　87881295

传真：(029)87880865